Computers Math. Applic. Vol. 27, No. 3, pp. 59–90, 1994
Printed in Great Britain.

0898-1221/94 $6.00 + 0.00

Kernel Optimum Nearly-Analytical Discretization (KOND) Algorithm Applied to Parabolic and Hyperbolic Equations

Y. KONDOH,* Y. HOSAKA AND K. ISHII†
Department of Electronic Engineering, Gunma University
Kiryu, Gunma 376, Japan

(Received October 1992; revised and accepted May 1993)

Abstract—Two applications of the Kernel Optimum Nearly-analytical Discretization (KOND) algorithm to the parabolic and the hyperbolic type equations are presented in detail to lead to novel numerical schemes with very high numerical accuracy. It is demonstrated numerically that the two-dimensional KOND-P scheme for the parabolic type yields much less numerical error by over 2–3 orders, measured quantitatively by the root mean square deviation from analytical solutions, and reduces the CPU time to about 1/5 for a common numerical accuracy, compared with the conventional explicit scheme of reference. It is also demonstrated numerically that the KOND-H scheme for the hyperbolic type yields much less diffusive error and has fairly high stability for both of the linear and the nonlinear wave propagations compared with other conventional schemes. Origins of numerical errors in data processing in general numerical schemes are discussed.

1. INTRODUCTION

Various numerical algorithms for solving the three types of partial differential problems of hyperbolic, elliptic, and parabolic equations have been developed [1–18]. While they yield fairly good results for many problems, more effort will be required to attain higher accuracy and stability when we investigate further the finer structure of the problem being studied. One of the authors (Y. K.) has reported a thought analysis on numerical schemes and developed a new algorithm called "Kernel Optimum Nearly-analytical Discretization (KOND) algorithm" for the construction of numerical schemes [19]. In the thought analysis, we investigate logical structures, ideas, or thoughts used in the objects being studied, and try to find some key elements for improvement and/or some other new thoughts which involve generality [19–23]. In the previous two reports [19,24], preliminary numerical results have been shown for two novel numerical schemes of the 1-D 2^{nd} KOND-H scheme and the 1-D 1^{st} KOND-P scheme, which are two applications of the KOND algorithm, respectively, to the one-dimensional hyperbolic type equation to the 2^{nd} derivatives and the one-dimensional parabolic type equation to the 1^{st} derivatives. It has been demonstrated by the numerical results of the 1-D 2^{nd} KOND-H scheme that the KOND-H scheme yields much

*Author to whom all correspondence is to be addressed.
†The authors would like to thank T. Yabe, J. L. Liang at Gunma University and R. Horiuchi at the National Institute for Fusion Science for their valuable discussion on numerical schemes and simulation methods. They also appreciate T. Yumoto, M. Yoshizawa, and M. Suzui for assistance with the computer programming and the preparation of the manuscript. One of the authors (Y. K.) thanks M. Plastow at NHK and T. Sato at the National Institute for Fusion Science for their contributions to this paper through constructive discussion on the thought analysis for the method of science. The authors would like to thank the referee for calling their attention to the problems possessing discontinuities.

This work was carried out under the collaborative research program at the National Institute for Fusion Science, Nagoya, Japan.

less diffusive error compared with other conventional schemes and has fairly high stability [19]. It has been also demonstrated by the numerical results that there appears higher diffusive error and/or noise in the calculation of the higher derivatives of the solutions. This structural property of the error would be common in all numerical schemes. The numerical results of the 1-D 1st KOND-P scheme has been shown to demonstrate much less numerical error than those by the conventional explicit scheme by 2–3 orders, measured quantitatively by the root mean square deviation from analytical solutions [24].

In this paper, we present in detail the applications of the KOND algorithm to the parabolic type equation and the hyperbolic type one. In the case of the parabolic type equation, we present two schemes for the one-dimensional and the two-dimensional equations and show typical numerical results demonstrating that the 2-D 1st KOND-P scheme yields much less numerical error by over 2–3 orders and reduces the CPU time to about 1/5 to attain the same common numerical accuracy, compared with the conventional explicit scheme of reference. In the case of the hyperbolic type equation, we present two types of the 1-D 2nd and the 1-D 1st KOND-H schemes, and show typical numerical results demonstrating high numerical accuracy of these two schemes, compared with the compact CIP (Cubic Interpolated Pseudo-particle) scheme [15–18], which is known to be a less diffusive scheme compared with other conventional schemes.

In Section 2, the KOND algorithm deduced from the thought analysis on numerical schemes is shown briefly together with the concept of losses of two types of information on "relations" embedded in differential equations and on functional "values" in solutions. Applications of the KOND algorithm to the parabolic type equation are presented in Section 3. In Section 3.1, we present the detailed procedure for the development of a scheme to obtain the discrete solutions to the 1st derivatives for the one-dimensional parabolic equation, in order to show the basic processes of the KOND algorithm and the origin of the resultant numerical accuracy. The two-dimensional parabolic equation is treated in Section 3.2, in order to show an example for effective high reduction of the CPU time to attain the same numerical accuracy. Applications of the KOND algorithm to the hyperbolic type equation are shown in Section 4. In Section 4.1, we present the KOND-H scheme with very high numerical accuracy which solves the one-dimensional hyperbolic equation to get discrete solutions up to the 2nd derivatives. In Section 4.2, we present simpler KOND-H schemes with fairly high numerical accuracy which solve the one-dimensional hyperbolic equation to get discrete solutions up to the 1st derivatives. Comparisons among numerical results by the KOND-H schemes and by the compact CIP scheme are presented in Section 4.3. Discussion and summary are given in Section 5. In the KOND algorithm shown in this paper, we confine ourselves to problems possessing the requisite level of continuity. Treatment of problems possessing discontinuities is discussed briefly in Section 5.

2. KOND ALGORITHM DEDUCED FROM THOUGHT ANALYSIS ON NUMERICAL SCHEMES

We present here briefly the thought analysis on numerical schemes to lead to the KOND algorithm [19]. In order to understand the structure of the ideas or thoughts used for numerical schemes for simulation, we try to analyze the basic process for solving a partial differential problem,

$$Lf(\mathbf{x}) = g(\mathbf{x}), \quad \text{for } \mathbf{x} = (x_1, \ldots, x_d) \text{ in a domain } \Omega \subset \mathbb{R}^d, \tag{1}$$

where L is a linear or nonlinear differential operator, $f(\mathbf{x})$ is an unknown function, and $g(\mathbf{x})$ is a given function. When it is hard to solve equation (1) analytically, we use usually two approximate methods, i.e., one is the approximate analytic method such as the perturbation method and the other is the discretization of equation (1) to solve the finite-difference equations. When we compare the ideas or thoughts themselves involved in the two methods, we may find the following elements of thoughts (we call the idea or thought itself involved in some method, simply "thought [A]").

In the approximate analytic method:

[A] to find global approximate continuous solutions.

[B] to find local approximate continuous solutions.

In the discretization method:

[C] to find finite-difference equations approximately equal to source equations.

[D] to find discrete approximate solutions on grids.

When we consider the processes for solving the source equation (1) to obtain its solution $f(\mathbf{x})$, we notice that there exist the following two types of information, [Inf.1] and [Inf.2], in the whole system of the source equation and its solution, and we use the two types of information in the data processing to obtain the solution:

[Inf.1] information on the "relations" which are embedded in the source equation, equation (1) and connecting the local values and their time evolutions.

[Inf.2] information on the functional "values" embedded in the solution $f(\mathbf{x})$.

Since the finite-difference equation for equation (1) itself has finite error compared with the source equation, equation (1), and therefore, it has finite loss of the information of [Inf.1] on the "relations" mentioned above, we had better solve equation (1) as directly as possible, avoiding use of the finite-difference equation.

We assume here that the analytic true solution $f(\mathbf{x})$ of equation (1) is obtained. The whole information of [Inf.2], mentioned above, that gives the whole property or character of equation (1) and its solution is included in the following set on the functional "values" of the analytic solution and its derivatives

$$\{f(\mathbf{x}),\ \partial_i f(\mathbf{x}),\ \partial_{ij} f(\mathbf{x}), \ldots\}, \tag{2}$$

where $\partial_i f(\mathbf{x})$, $\partial_{ij} f(\mathbf{x}), \ldots$ stand, respectively, for $\dfrac{\partial f(\mathbf{x})}{\partial x_i}$, $\dfrac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}$, and so on. Each element of the set of equation (2) obeys, respectively, the following set on the "relations" of differential equations

$$\{\text{eq.}(1),\ \text{eq.}(4),\ \text{eq.}(5),\ \ldots\}, \tag{3}$$

where equations (4), (5),..., are the following:

$$\partial_i\,[Lf(\mathbf{x}) = g(\mathbf{x})], \qquad \text{for } \mathbf{x} = (x_1, \ldots, x_d), \tag{4}$$

$$\partial_{ij}\,[Lf(\mathbf{x}) = g(\mathbf{x})], \qquad \text{for } \mathbf{x} = (x_1, \ldots, x_d), \tag{5}$$

.....

We call equation (1) "the source equation," equation (4) "the first branch equations," equation (5) "the second branch equations," and so on. These source and branch equations include the important information of [Inf.1] on the "relations" mentioned above.

We first analyze the information of [Inf.2] on the functional "values." Mapping the set on the functional "values" of the analytic solutions, equation (2), onto the grid points $\mathbf{x}_n^h$ in a given uniform or nonuniform grid G^h with mesh size $\mathbf{h}_d$, we may obtain the following set of {discrete values of solutions at grid points, interpolation curves around grid points, connection relations at neighboring grid points} which is equivalent to the set of equation (2):

⟨*set of discrete values of solutions at grid points*⟩

$$\{f_n,\ \partial_i f_n,\ \partial_{ij} f_n, \ldots\}, \tag{6}$$

where the subscript n denotes here a d-dimensional integer.

⟨*set of interpolation curves around grid points*⟩

$$\{F_n(\mathbf{s}),\ \partial_i F_n(\mathbf{s}),\ \partial_{ij} F_n(\mathbf{s}), \ldots\}, \tag{7}$$

where $\mathbf{s}$ is defined as $\mathbf{s} \equiv \mathbf{x} - \mathbf{x}_n^h$.

⟨*set of connection relations at neighboring grid points*⟩

$$F_n(-\mathbf{h}_d) = f_{n-1}, \qquad F_n(\mathbf{h}_d) = f_{n+1}, \tag{8}$$

$$\partial_i F_n(-\mathbf{h}_d) = \partial_i f_{n-1}, \qquad \partial_i F_n(\mathbf{h}_d) = \partial_i f_{n+1}, \tag{9}$$

$$\partial_{ij} F_n(-\mathbf{h}_d) = \partial_{ij} f_{n-1}, \qquad \partial_{ij} F_n(\mathbf{h}_d) = \partial_{ij} f_{n+1}. \tag{10}$$

.....

Each element of the set, equation (7), should be the piecewise segment of the corresponding analytic solutions of equation (2). The set of {the discrete solutions equation (6), the segmental interpolation curves equation (7), the connection relations of equations (8), (9), (10), ...} is exactly equivalent to the set of the true solutions, equation (2). Using the Taylor expansion, the elements of the set of the interpolation curves, equation (7), can be written as follows:

$$F_n(\mathbf{s}) = f_n + \sum_i \partial_i f_n s_i + \sum_{i,j} \partial_{ij} f_n s_i s_j / 2 + \cdots, \tag{11}$$

$$\partial_i F_n(\mathbf{s}) = \partial_i f_n + \sum_j \partial_{ij} f_n s_j + \sum_{j,k} \partial_{ijk} f_n s_j s_k / 2 + \cdots, \tag{12}$$

$$\partial_{ij} F_n(\mathbf{s}) = \partial_{ij} f_n + \sum_k \partial_{ijk} f_n s_k + \sum_{k,l} \partial_{ijkl} f_n s_k s_l / 2 + \cdots, \tag{13}$$

.....

We see from comparison between equation (6) and equations (11), (12), (13),... that the discrete values of solution, equation (6), are themselves the coefficients of the interpolation curves by the Taylor expansion and, therefore, they can induce good approximate and locally continuous solutions around the grid points. In other words, the infinite set of the discrete values of equation (6) itself becomes one of the best discretizations for the whole information on the functional "values" of the continuous true solutions, equation (2), based upon the interpolation curves by the Taylor expansions. This corresponds analogically to the representation of a given function by the discrete spectra with use of the Fourier expansion. Since we cannot use the infinite elements of the set of equation (6), we use two or three elements from the beginning in equation (6), for example, f_n, $\partial_i f_n$, $\partial_{ij} f_n$. We then the lose finer information included in the rest of the infinite terms beyond the terms of $\partial_{ij} f_n$, in this example, in the Taylor expansions. The rest of the infinite terms are considered to carry the semiglobal information for the uncovered regions between the neighboring grid points that make the interpolation curves satisfy the connection relations of equations (8)–(10). Using reversely the connection relations and introducing additional Taylor coefficients, i.e., three more additional terms for the one-dimensional problem in this example, we can recover approximately the lost information in the rest of the infinite terms by the additional terms. In other words, the rest of the infinite terms in the Taylor expansions can be folded up approximately in the finite additional terms by using the connection relations. From the thought analysis on the loss of [Inf.2] shown above, we find that if we use more elements in the set of equation (6) and if we use more connection relations for the adopted elements in order to recover approximately the lost information in the rest of the infinite terms in the Taylor expansions by folding up approximately the rest of the infinite terms into some additional Taylor terms, then we can suppress effectively the loss of [Inf.2] in data processing.

We next analyze the information of [Inf.1] on the "relations" in the differential equations. In order to obtain the values of the adopted elements of equation (6), we can use the source equation

and the branch equations corresponding to the adopted elements, and all of those equations would have a common type of differential equations with each other. In order to suppress the loss of [Inf.1], we should use more elements in the set of equation (3) with respect to the source and its branch equations. It is because if we use branch equations up to the higher order, we can embed correct information on the "relations" carried by the branch equations into the data processing up to the corresponding higher order derivatives, as will be shown in examples in the next section. On the other hand, when we solve the source and the branch equations with use of the conventional finite-difference equations, we cannot avoid loss of [Inf.1] on the "relations" by the discretization itself of the source and the branch equations. In order to suppress the loss of [Inf.1] by the discretization of the source and the branch equations, we should find higher-order approximate analytic solutions for the source and the branch equations as analytically as possible.

We notice from the above analysis on the losses of [Inf.1] and [Inf.2] that if we have a method which is nearly analytical for obtaining better approximate solutions for the more elements of the set of equation (6), we would get the more accurate and denser information for the set of the true solution, equation (2). The accuracy of the information for the solutions by this method is optimum at the grid points, as is seen from the above argument, and in other words, the discretization by this method is kernel optimum.

The thought analysis mentioned above leads to the following main set of thoughts to be used in the algorithm for the construction of the numerical scheme, which is the combination of the elements of the two sets of thoughts for the approximate analytic method {[A],[B]} and the discretization method {[C],[D]}. We call the algorithm "Kernel Optimum Nearly-analytical Discretization (KOND) algorithm."

The main set of thoughts of the KOND algorithm is

$$\{[\,\mathrm{I}\,],\ [\mathrm{II}],\ [\mathrm{III}],\ [\mathrm{IV}]\}, \tag{14}$$

where the four elements of thought are as follows:

[I] to use the source equations and their branch equations {Equations (1), (4), (5), ...} as much as possible.

[II] to find higher-order approximate analytic solutions as analytically as possible, by using some methods such as the perturbation method, the Taylor expansion, and others.

[III] to find the set of discrete solutions $\{f_n,\ \partial_i f_n,\ \partial_{ij} f_n, \ldots\}$, as many elements as possible, that are the coefficients of the interpolation curves {Equations (11), (12), (13), ...} by the Taylor expansions corresponding to the local continuous solutions around the grid points. ["The kernel optimum discretization of a function by the coefficients of the Taylor expansion at every grid point."]

[IV] to use the set of the connection relations {Equations (8), (9), (10),...}, as many elements as possible, in order to include the semiglobal information for the uncovered regions between the neighboring gird points and to find the additional higher order Taylor coefficients which represent approximately the rest of the infinite terms of the Taylor expansion. ["The folding up of the rest of the terms of the Taylor expansion by the connection relations."]

The two thoughts of {[I], [II]} are essential to attain higher numerical accuracy by suppressing the losses of [Inf.1] on the "relations" embedded in the source and the branch differential equations. The other two thoughts of {[III], [IV]} are essential to attain further higher numerical accuracy by suppressing the losses of [Inf.2] on the functional "values" of $f(\mathbf{x})$. Each of these four thoughts {[I], [II], [III], [IV]} of the KOND algorithm may seem to be rather simple and abstract to develop new schemes with higher numerical accuracy. In the following

sections, however, the set of the four thoughts of the KOND algorithm will be shown to give us novel numerical schemes which yield quite high accuracy and, therefore, effective high reduction of the CPU time to attain the same numerical accuracy.

3. KOND ALGORITHM FOR PARABOLIC EQUATIONS (KOND-P SCHEME)

We apply here the KOND algorithm to the numerical scheme for solving the parabolic type equation. In the following Section 3.1, we present the procedure for the development of a scheme to obtain the discrete solutions to the 1st derivatives, i.e., f_n and $\partial_x f_n$, for the one-dimensional parabolic equation in order to show the basic processes of the KOND algorithm and the origin of the resultant numerical accuracy. We express here the scheme for the one-dimensional parabolic equation to the 1st derivatives by the KOND algorithm as "1-D 1st KOND-P scheme." We use the notations such as $\partial_t f$, $\partial_x f$, and $\partial_{mx} f$ as the abbreviations for $\dfrac{\partial f(t,x)}{\partial t}$, $\dfrac{\partial f(t,x)}{\partial x}$, and $\dfrac{\partial^m f(t,x)}{\partial x^m}$, respectively. In the Section 3.2, we present a 2-D 1st KOND-P scheme, which is the scheme to obtain the discrete solutions to the 1st derivatives for the two-dimensional parabolic equation, in order to show an example for effective high reduction of the CPU time to attain the same numerical accuracy.

3.1. One-Dimensional 1st Order Case (1-D 1st KOND-P Scheme)

We treat here a one-dimensional parabolic equation used for diffusion equations, and develop a scheme to obtain the discrete solutions to the 1st derivatives. According to the first thought element, [I], of equation (14), and from equation (4), we solve the source equation and the 1st branch equation for the one-dimensional diffusion equation, which are written, respectively, as follows:

$$\partial_t f(t,x) = P, \tag{15}$$

$$\partial_{tx} f = \partial_x P, \tag{16}$$

$$P \equiv \partial_x [D(t,x)\partial_x f(t,x)], \tag{17}$$

where $D(t,x)$ in the definition of P of equation (17) is a given diffusion coefficient. The m^{th} branch equation for the source equation, equation (15), is written generally as

$$\partial_{tmx} f = \partial_{mx} P. \tag{18}$$

We use the higher order branch equations of equation (18) as much as possible in the following procedure because of the suppression of the losses of [Inf.1] on the "relations" in the differential equations, as was discussed in the previous section.

According to the second thought element, [II], of equation (14), we solve equations (15)–(17) locally around a point of (t_k, x_n) as analytically as possible. Using the Taylor expansion around the time of t_k, we write $\partial_{mx} P_n$ as

$$\partial_{mx} P_n = \partial_{mx} P_n^k + \partial_{tmx} P_n^k \tau + \cdots, \tag{19}$$

where $\tau \equiv t - t_k$, $\partial_{mx} P_n \equiv \partial_{mx} P(t, x_n)$, and $\partial_{mx} P_n^k \equiv \partial_{mx} P(t_k, x_n)$. When $m = 0$ in equation (19), then equation (19) becomes the Taylor expansion for P itself. Using equation (19) and integrating equations (15) and (16) with respect to τ over the time interval of Δt, we obtain approximate solutions for f_n^{k+1} and $\partial_x f_n^{k+1}$ at the point of (t_{k+1}, x_n) as follows:

$$f_n^{k+1} = f_n^k + P_n^k \Delta t + \frac{1}{2}\partial_t P_n^k (\Delta t)^2 + \cdots, \tag{20}$$

$$\partial_x f_n^{k+1} = \partial_x f_n^k + \partial_x P_n^k \Delta t + \frac{1}{2}\partial_{tx} P_n^k (\Delta t)^2 + \cdots, \tag{21}$$

where $f_n^k \equiv f(t_k, x_n)$ and $\partial_x f_n^k \equiv \partial_x f(t_k, x_n)$. The first order approximate solution of equation (20) with respect to Δt yields the ordinary finite-difference equation. This finite-difference equation loses much information of [Inf.1] on the "relations" in the given differential equations corresponding to the higher order terms in equation (20) and also equation (21) and higher order branch equations. We use here the second order approximate analytic solutions with respect to Δt in equations (20) and (21), for simplicity.

We now proceed to the third thought element, [III], of equation (14). In order to find the set of discrete solutions of $\{f_n^{k+1},\ \partial_x f_n^{k+1}\}$ with use of equations (20) and (21), we have to express the right hand sides of equations (20) and (21) by the values at the time of t_k and/or t_{k-1}. Using the definition of equation (17), we obtain $P_n^k, \partial_t P_n^k, \partial_x P_n^k$ and $\partial_{tx} P_n^k$ in equations (20) and (21) as follows:

$$P_n^k = \partial_x D_n^k \partial_x f_n^k + D_n^k \partial_{2x} f_n^k, \tag{22}$$

$$\partial_t P_n^k = \partial_{tx} D_n^k \partial_x f_n^k + \partial_x D_n^k \partial_{tx} f_n^k + \partial_t D_n^k \partial_{2x} f_n^k + D_n^k \partial_{t2x} f_n^k, \tag{23}$$

$$\partial_x P_n^k = \partial_{2x} D_n^k \partial_x f_n^k + 2\partial_x D_n^k \partial_{2x} f_n^k + D_n^k \partial_{3x} f_n^k, \tag{24}$$

$$\begin{aligned}\partial_{tx} P_n^k = {} & \partial_{t2x} D_n^k \partial_x f_n^k + \partial_{2x} D_n^k \partial_{tx} f_n^k + 2\partial_{tx} D_n^k \partial_{2x} f_n^k + 2\partial_x D_n^k \partial_{t2x} f_n^k \\ & + \partial_t D_n^k \partial_{3x} f_n^k + D_n^k \partial_{t3x} f_n^k.\end{aligned} \tag{25}$$

In equations (23) and (25), there appear again the terms of $\partial_{tx} f_n^k$, $\partial_{t2x} f_n^k$ and $\partial_{t3x} f_n^k$ to be determined. We can use again the branch equations, equation (18), to the third order for the determination of these terms as follows:

$$\begin{aligned}\partial_{tx} f_n^k &= \partial_x P_n^k \\ &= \partial_{2x} D_n^k \partial_x f_n^k + 2\partial_x D_n^k \partial_{2x} f_n^k + D_n^k \partial_{3x} f_n^k,\end{aligned} \tag{26}$$

$$\begin{aligned}\partial_{t2x} f_n^k &= \partial_{2x} P_n^k \\ &= \partial_{3x} D_n^k \partial_x f_n^k + 3\partial_{2x} D_n^k \partial_{2x} f_n^k + 3\partial_x D_n^k \partial_{3x} f_n^k + D_n^k \partial_{4x} f_n^k,\end{aligned} \tag{27}$$

$$\begin{aligned}\partial_{t3x} f_n^k &= \partial_{3x} P_n^k \\ &= \partial_{4x} D_n^k \partial_x f_n^k + 4\partial_{3x} D_n^k \partial_{2x} f_n^k + 6\partial_{2x} D_n^k \partial_{3x} f_n^k + 4\partial_x D_n^k \partial_{4x} f_n^k + D_n^k \partial_{5x} f_n^k.\end{aligned} \tag{28}$$

It should be emphasized here again that when we use the branch equations to the higher order, we can obtain the higher numerical accuracy. This is because by using the branch equations as much as possible, we can avoid the loss of the important information of [Inf.1] on the "relations" to the higher derivatives, which are embedded in the source and branch equations and connecting the local values and their time evolutions.

Using the values at the time of t_k and/or t_{k-1}, we obtain the derivatives of D, for example, $\partial_t D_n^k = \left(D_n^k - D_n^{k-1}\right)/\Delta t$, $\partial_x D_n^k = \left(D_{n+1}^k - D_{n-1}^k\right)/2h$, and so on, where $h\,(= x_n - x_{n-1})$ is the mesh size. When we use the approximate solutions for f_n^{k+1} and $\partial_x f_n^{k+1}$ to the order of $(\Delta t)^2$ in equations (20) and (21), we can determine the values of f_n^{k+1} and $\partial_x f_n^{k+1}$ with use of equations (22)–(28), the values of D at the time of t_k and t_{k-1}, and the values of $\partial_{mx} f_n^k$ $(m = 0, 1, 2, 3, 4, 5)$. (If we neglect four terms of $\partial_{mx} f_n^k$ $(m = 2, 3, 4, 5)$ in equations (22)–(28) as an approximation, then we can still obtain the values of f_n^{k+1} and $\partial_x f_n^{k+1}$ without using any additional thought element. In this sense, we are free from using the connection relations of equations (8)–(10) in the fourth thought element, [IV], of equation (14). In order to attain higher numerical accuracy, we need the fourth thought element [IV].)

We proceed to the final thought element, [IV], of equation (14). Since we have to determine the values of f_n^{k+1} and $\partial_x f_n^{k+1}$ with higher numerical accuracy from those of f_n^k and $\partial_x f_n^k$, we need the values of $\partial_{mx} f_n^k$ $(m = 2, 3, 4, 5)$ included in equations (22)–(28). We therefore use the following interpolation curves up to the term of $\partial_{5x} f_n^k$ of the Taylor expansion, which represent approximately the rest of the infinite terms of the Taylor expansion,

$$F_n^k(s) = f_n^k + \partial_x f_n^k s + \frac{1}{2}\partial_{2x} f_n^k s^2 + \frac{1}{6}\partial_{3x} f_n^k s^3 + \frac{1}{24}\partial_{4x} f_n^k s^4 + \frac{1}{120}\partial_{5x} f_n^k s^5, \tag{29}$$

$$\partial_x F_n^k(s) = \partial_x f_n^k + \partial_{2x} f_n^k s + \frac{1}{2}\partial_{3x} f_n^k s^2 + \frac{1}{6}\partial_{4x} f_n^k s^3 + \frac{1}{24}\partial_{5x} f_n^k s^4, \tag{30}$$

where $s \equiv x - x_n$. We use following four connection relations for f_n^k and $\partial_x f_n^k$ from equations (8) and (9), in order to fold up the information included in the rest of the infinite terms of the Taylor expansion into the additional four terms of $\partial_{mx} f_n^k$ $(m = 2, 3, 4, 5)$ and also to determine their values by f_n^k and $\partial_x f_n^k$,

$$F_n^k(-h) = f_{n-1}^k, \tag{31}$$

$$F_n^k(h) = f_{n+1}^k, \tag{32}$$

$$\partial_x F_n^k(-h) = \partial_x f_{n-1}^k, \tag{33}$$

$$\partial_x F_n^k(h) = \partial_x f_{n+1}^k. \tag{34}$$

Substituting equations (29) and (30) into equations (31)–(34), we obtain the four additional Taylor coefficients, $\partial_{mx} f_n^k$ $(m = 2, 3, 4, 5)$, which are given by f_n^k and $\partial_x f_n^k$, as follows:

$$\partial_{2x} f_n^k = \frac{2}{h^2}\left(f_{n+1}^k - 2f_n^k + f_{n-1}^k\right) - \frac{1}{2h}\left(\partial_x f_{n+1}^k - \partial_x f_{n-1}^k\right), \tag{35}$$

$$\partial_{3x} f_n^k = \frac{15}{2h^3}\left(f_{n+1}^k - f_{n-1}^k\right) - \frac{3}{2h^2}\left(\partial_x f_{n+1}^k + 8\partial_x f_n^k + \partial_x f_{n-1}^k\right), \tag{36}$$

$$\partial_{4x} f_n^k = -\frac{12}{h^4}\left(f_{n+1}^k - 2f_n^k + f_{n-1}^k\right) + \frac{6}{h^3}\left(\partial_x f_{n+1}^k - \partial_x f_{n-1}^k\right), \tag{37}$$

$$\partial_{5x} f_n^k = -\frac{90}{h^5}\left(f_{n+1}^k - f_{n-1}^k\right) + \frac{30}{h^4}\left(\partial_x f_{n+1}^k + 4\partial_x f_n^k + \partial_x f_{n-1}^k\right). \tag{38}$$

Using the initial values of f_n, i.e., f_n^1, we obtain the initial values of $\partial_x f_n$, i.e., $\partial_x f_n^1$, by $\partial_x f_n^1 = \left(f_{n+1}^1 - f_{n-1}^1\right)/2h$. We may notice from equations (21)–(38) shown above that we avoid the losses of [Inf.2] on the functional "values" in $f(t, x)$ by finding the values of $\partial_{mx} f_n^k$ $(m = 1, 2, 3, 4, 5)$ and folding up the information included in the rest of the infinite terms of the Taylor expansion into the additional four terms of $\partial_{mx} f_n^k$ $(m = 2, 3, 4, 5)$. If we treat only the values of f_n^k instead of $\left\{f_n^k,\ \partial_x f_n^k\right\}$, we do lose the important information of [Inf.2] on the functional "values" shown above, and these losses of [Inf.2] will accumulate numerical error to increase during data processing.

We now consider how to treat the boundary values of f_n^k and $\partial_x f_n^k$. The boundary conditions for the source equation (15) usually are given in one of the following two forms:

$$\text{boundary condition (a):} \qquad f_n^k = \text{ const. } (n = 1,\ N), \tag{39}$$

$$\text{boundary condition (b):} \qquad \partial_x f_n^k = \text{ const. } (n = 1, N), \tag{40}$$

where $(n = 1$ and $n = N)$ denote the boundary grids. We show here how to determine the values of $\partial_x f_n^k$ $(n = 1, N)$ for the case of the boundary condition (a) of equation (39). [If we use the boundary condition (b) of equation (40), then we exchange f_1^k with $\partial_x f_1^k$ in the following argument.] According to the thought elements [III] and [IV] of equation (14), the interpolation curve around the grid point x_2 and the connection relations are given, respectively, from equations (29) and (30) and equations (31)–(34) as follows:

$$F_2^k(s) = f_2^k + \partial_x f_2^k s + \frac{1}{2}\partial_{2x} f_2^k s^2 + \frac{1}{6}\partial_{3x} f_2^k s^3 + \frac{1}{24}\partial_{4x} f_2^k s^4 + \frac{1}{120}\partial_{5x} f_2^k s^5, \tag{41}$$

$$\partial_x F_2^k(s) = \partial_x f_2^k + \partial_{2x} f_2^k s + \frac{1}{2}\partial_{3x} f_2^k s^2 + \frac{1}{6}\partial_{4x} f_2^k s^3 + \frac{1}{24}\partial_{5x} f_2^k s^4, \tag{42}$$

$$F_2^k(-h) = f_1^k, \tag{43}$$

$$F_2^k(h) = f_3^k, \tag{44}$$

$$\partial_x F_2^k(-h) = \partial_x f_1^k, \tag{45}$$

$$\partial_x F_2^k(h) = \partial_x f_3^k. \tag{46}$$

Since $\partial_x f_1^k$ is unknown this time in addition to $\partial_{mx} f_2^k$ $(m = 2, 3, 4, 5)$, we have to remove the last term of $\partial_{5x} f_2^k$ in equations (41) and (42). Substituting equations (41) and (42) without the term of $\partial_{5x} f_2^k$ into equations (43)–(46), we obtain the three additional Taylor coefficients, $\partial_{mx} f_2^k$ $(m = 2, 3, 4)$, and $\partial_x f_1^k$, as follows:

$$\partial_{2x} f_2^k = \frac{1}{2h^2}\left(7f_3^k - 8f_2^k - f_1^k\right) - \frac{1}{h}\left(\partial_x f_3^k + 2\,\partial_x f_2^k\right), \tag{47}$$

$$\partial_{3x} f_2^k = \frac{3}{h^3}\left(f_3^k - f_1^k\right) - \frac{6}{h^2}\,\partial_x f_2^k, \tag{48}$$

$$\partial_{4x} f_2^k = \frac{6}{h^4}\left(-5f_3^k + 4f_2^k + f_1^k\right) + \frac{12}{h^3}\left(\partial_x f_3^k + 2\,\partial_x f_2^k\right), \tag{49}$$

$$\partial_x f_1^k = \partial_x f_2^k - \partial_{2x} f_2^k h + \frac{1}{2}\partial_{3x} f_2^k h^2 - \frac{1}{6}\partial_{4x} f_2^k h^3. \tag{50}$$

Substituting equations (47)–(49) into equation (50), we can determine the value of $\partial_x f_1^k$ from the values of f_n^k $(n = 1, 2, 3)$ and $\partial_x f_n^k$ $(n = 2, 3)$. Using the same process mentioned above, we can determine the value of $\partial_x f_N^k$ by replacing h and the subscripts $\{1, 2, 3\}$ for grid points in equations (47)–(50) with $-h$ and $\{N, N-1, N-2\}$, respectively.

Combining all processes for the four thoughts $\{[\,\mathrm{I}\,], [\mathrm{II}], [\mathrm{III}], [\mathrm{IV}]\}$ and for the boundary values, i.e., using equations (35)–(38), equations (22)–(28), equations (20) and (21) to the order of $(\Delta t)^2$, and also equations (47)–(50), we find the set of the discrete solutions to the 1st derivatives $\left\{f_n^{k+1}, \partial_x f_n^{k+1}\right\}$ after one time step from the state of $\left\{f_n^k, \partial_x f_n^k\right\}$.

We show here some typical numerical results by the 1-D 1st KOND-P scheme shown above, comparing with the results by the conventional explicit scheme as a reference measure. In order to test the accuracy of the numerical results, we calculate the following case with the analytical solution: the diffusion coefficient $D = 1$, the initial profile of $f(0, x) = \sin(2\pi x/\lambda)$, and the boundary condition (a) of $f_n^k = 0$ $(n = 1, N)$, where λ is one period length of $f(x)$. The analytical solution for this case is written as $f(t, x) = \exp\left[-\left(\frac{2\pi}{\lambda}\right)^2 t\right] \sin(2\pi x/\lambda)$. When we define M as the number of meshes in one period length, λ is given by $\lambda = Mh$, and $M + 1$ grid points cover one period length. We use the following conventional explicit scheme, denoted by "1-D EXPL" hereafter, obtained from the finite-difference equation, for comparison: $f_n^{k+1} = f_n^k + D\Delta t\left(f_{n+1}^k - 2f_n^k + f_{n-1}^k\right)/h^2$. In order to measure quantitatively the numerical accuracy, we use the root mean square deviation, σ, from the analytical solution, which is defined by

$$\sigma = \left\{\frac{1}{N}\sum_{n=1}^{N}\left[f_n^k - f(t_k, x_n)\right]^2\right\}^{1/2}. \tag{51}$$

The double precision programme is used for the following numerical calculations.

Figure 1 shows typical results of computation for the time evolution of σ in the case of $M = 20$ and $D\Delta t/h^2 = 0.1$, where two lines of σ for the 1-D EXPL scheme (the mark □) and the 1-D 1st KOND-P scheme (the mark ■) are shown in a semi-log scale. We recognize from Figure 1 that the error of the 1-D 1st KOND-P scheme measured by σ is less than that of the 1-D EXPL one by about 2 orders in this case. It is also seen from Figure 1 that the rate of increment of σ in the 1-D 1st KOND-P scheme is less than that in the 1-D EXPL one.

Figure 2 shows typical results of computation to see the dependence of σ on the number of meshes M in one period length in the case of $D\Delta t/h^2 = 0.1$, where two lines of σ at the time of $t = 1.0$ are shown in a semi-log scale for the 1-D EXPL scheme (the mark □) and the 1-D 1st KOND-P scheme (the mark ■). It is recognized from Figure 2 that higher improvement rate of accuracy by increasing the number of meshes M can be achieved in the 1-D 1st KOND-P scheme than in the 1-D EXPL one. In the case of $M = 40$, the error of the 1-D 1st KOND-P scheme measured by σ becomes less than that of the 1-D EXPL one by about 3 orders, as is seen in Figure 2. The data in Figure 2 also shows that improvement of accuracy by increasing the number of meshes saturates faster in the 1-D EXPL scheme than in the 1-D 1st KOND-P one.

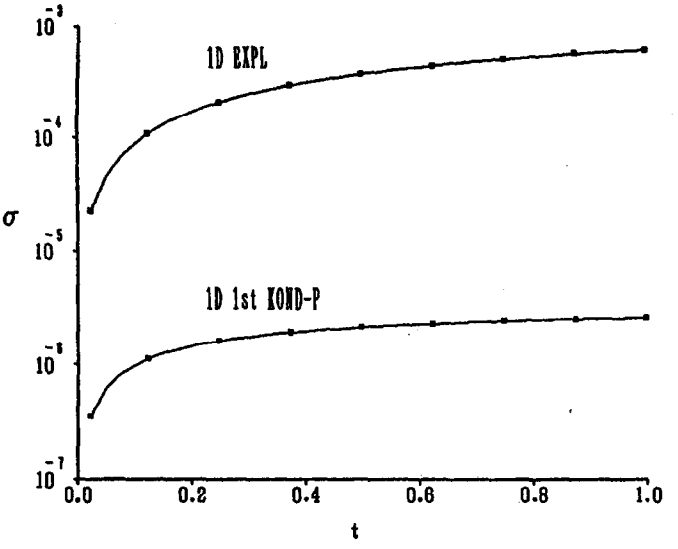

Figure 1. Typical results of computation for the time evolution of numerical error measured by σ in the case of $M = 20$ and $D\Delta t/h^2 = 0.1$. Two lines of σ for the 1-D EXPL scheme (the mark □) and the 1-D 1st KOND-P scheme (the mark ■) are shown in a semi-log scale.

We recognize from Figures 1 and 2 that quite high accuracy can be attained by the present 1-D 1st KOND-P scheme. The local multisubscales and delta function (LMS-DF) method reported in [25] to improve numerical schemes is also applicable to the present 1-D 1st KOND-P scheme as well as to the 1-D 2nd KOND-H scheme [19] for the hyperbolic equation to attain even less numerical error.

3.2. Two-Dimensional 1st Order Case (2-D 1st KOND-P Scheme)

We treat here a two-dimensional parabolic equation used for diffusion equations, and develop a scheme to obtain the discrete solutions to the 1st derivatives. According to the first thought element, [I], of equation (14), and from equation (4), the source equation and the 1st branch equations for the two-dimensional diffusion equation are written, respectively, as follows:

$$\partial_t f(t,x,y) = P, \tag{52}$$

$$\partial_{tx} f = \partial_x P, \qquad \partial_{ty} f = \partial_y P \tag{53}$$

$$P \equiv \partial_x \left[D(t,x,y)\, \partial_x f(t,x,y) \right] + \partial_y \left[D(t,x,y)\, \partial_y f(t,x,y) \right], \tag{54}$$

where $D(t,x,y)$ in the definition of P of equation (54) is a given diffusion coefficient. The $(m+n)^{\text{th}}$ branch equations for the source equation (52) are written as

$$\partial_{tmxny} f = \partial_{mxny} P, \tag{55}$$

where $\partial_{mxny} f$ is an abbreviation for $\dfrac{\partial^{m+n} f}{\partial x^m \partial y^n}$.

According to the second thought element [II] of equation (14), we solve equations (52)–(54) locally around a point of (t_k, x_i, y_j). Using the Taylor expansion around the time of t_k, we write $\partial_{mxny} P_{i,j}$ as

$$\partial_{mxny} P_{i,j} = \partial_{mxny} P^k_{i,j} + \partial_{tmxny} P^k_{i,j} \tau + \cdots, \tag{56}$$

where $\tau \equiv t - t_k$, $\partial_{mxny} P_{i,j} \equiv \partial_{mxny} P(t, x_i, y_j)$, and $\partial_{mxny} P^k_{i,j} \equiv \partial_{mxny} P(t_k, x_i, y_j)$. When $m = n = 0$ in equation (56), then equation (56) becomes the Taylor expansion with respect to time τ for P itself. Using equation (56) and integrating equations (52) and (53) with respect to τ

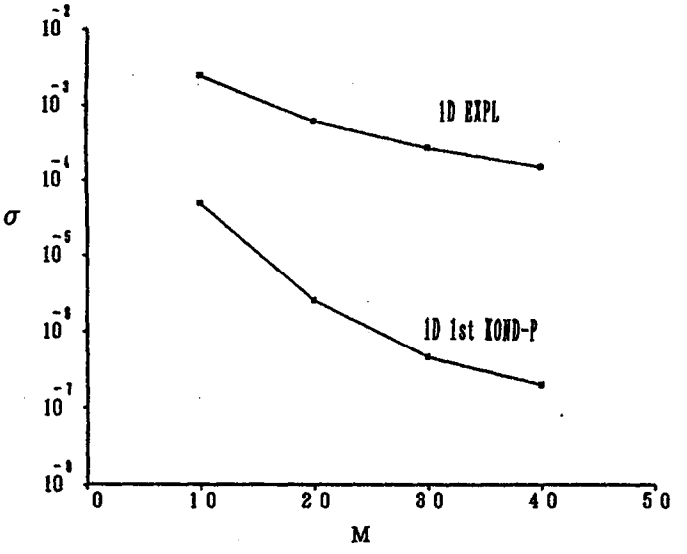

Figure 2. Dependence of numerical error measured by σ on the number of meshes M in one period length in the case of $D\Delta t/h^2 = 0.1$. Two lines of σ at the time of $t = 1.0$ for the 1-D EXPL scheme (the mark □) and the 1-D 1st KOND-P scheme (the mark ■) are shown in a semi-log scale.

over the time interval of Δt, we obtain approximate solutions for $f_{i,j}^{k+1}$, $\partial_x f_{i,j}^{k+1}$, and $\partial_y f_{i,j}^{k+1}$ at the point of (t_{k+1}, x_i, y_j) as follows:

$$f_{i,j}^{k+1} = f_{i,j}^k + P_{i,j}^k \Delta t + \frac{1}{2}\partial_t P_{i,j}^k (\Delta t)^2 + \cdots, \tag{57}$$

$$\partial_x f_{i,j}^{k+1} = \partial_x f_{i,j}^k + \partial_x P_{i,j}^k \Delta t + \frac{1}{2}\partial_{tx} P_{i,j}^k (\Delta t)^2 + \cdots, \tag{58}$$

$$\partial_y f_{i,j}^{k+1} = \partial_y f_{i,j}^k + \partial_y P_{i,j}^k \Delta t + \frac{1}{2}\partial_{ty} P_{i,j}^k (\Delta t)^2 + \cdots, \tag{59}$$

where $f_{i,j}^k \equiv f(t_k, x_i, y_j)$, $\partial_x f_{i,j}^k \equiv \partial_x f(t_k, x_i, y_j)$, and $\partial_y f_{i,j}^k \equiv \partial_y f(t_k, x_i, y_j)$. We use here the second order approximate analytic solutions with respect to Δt in equations (57)–(59).

We now proceed to the third thought element, [III], of equation (14). In order to find the set of discrete solutions of $\{f_{i,j}^{k+1}, \partial_x f_{i,j}^{k+1}, \partial_y f_{i,j}^{k+1}\}$ with use of equations (57)–(59), we have to express the right hand sides of equations (57)–(59) by the values at the time of t_k and/or t_{k-1}. Using the definition of equation (54), we obtain $P_{i,j}^k$, $\partial_t P_{i,j}^k$, $\partial_x P_{i,j}^k$, $\partial_{tx} P_{i,j}^k$, $\partial_y P_{i,j}^k$, and $\partial_{ty} P_{i,j}^k$ in equations (57)–(59) as follows:

$$P_{i,j}^k = \partial_x D_{i,j}^k \partial_x f_{i,j}^k + \partial_y D_{i,j}^k \partial_y f_{i,j}^k + D_{i,j}^k \left(\partial_{2x} f_{i,j}^k + \partial_{2y} f_{i,j}^k\right), \tag{60}$$

$$\begin{aligned}\partial_t P_{i,j}^k = {} & \partial_{tx} D_{i,j}^k \partial_x f_{i,j}^k + \partial_x D_{i,j}^k \partial_{tx} f_{i,j}^k + \partial_{ty} D_{i,j}^k \partial_y f_{i,j}^k + \partial_y D_{i,j}^k \partial_{ty} f_{i,j}^k \\ & + \partial_t D_{i,j}^k \left(\partial_{2x} f_{i,j}^k + \partial_{2y} f_{i,j}^k\right) + D_{i,j}^k \left(\partial_{t2x} f_{i,j}^k + \partial_{t2y} f_{i,j}^k\right),\end{aligned} \tag{61}$$

$$\begin{aligned}\partial_x P_{i,j}^k = {} & \partial_{2x} D_{i,j}^k \partial_x f_{i,j}^k + \partial_x D_{i,j}^k \partial_{2x} f_{i,j}^k + \partial_{xy} D_{i,j}^k \partial_y f_{i,j}^k + \partial_y D_{i,j}^k \partial_{xy} f_{i,j}^k \\ & + \partial_x D_{i,j}^k \left(\partial_{2x} f_{i,j}^k + \partial_{2y} f_{i,j}^k\right) + D_{i,j}^k \left(\partial_{3x} f_{i,j}^k + \partial_{x2y} f_{i,j}^k\right),\end{aligned} \tag{62}$$

$$\begin{aligned}\partial_{tx} P_{i,j}^k = {} & \partial_{t2x} D_{i,j}^k \partial_x f_{i,j}^k + \partial_{2x} D_{i,j}^k \partial_{tx} f_{i,j}^k + \partial_{tx} D_{i,j}^k \partial_{2x} f_{i,j}^k + \partial_x D_{i,j}^k \partial_{t2x} f_{i,j}^k \\ & + \partial_{txy} D_{i,j}^k \partial_y f_{i,j}^k + \partial_{xy} D_{i,j}^k \partial_{ty} f_{i,j}^k + \partial_{ty} D_{i,j}^k \partial_{2x} f_{i,j}^k + \partial_y D_{i,j}^k \partial_{txy} f_{i,j}^k \\ & + \partial_{tx} D_{i,j}^k \left(\partial_{2x} f_{i,j}^k + \partial_{2y} f_{i,j}^k\right) + \partial_x D_{i,j}^k \left(\partial_{t2x} f_{i,j}^k + \partial_{t2y} f_{i,j}^k\right) \\ & + \partial_t D_{i,j}^k \left(\partial_{3x} f_{i,j}^k + \partial_{x2y} f_{i,j}^k\right) + D_{i,j}^k \left(\partial_{t3x} f_{i,j}^k + \partial_{tx2y} f_{i,j}^k\right),\end{aligned} \tag{63}$$

$$\partial_y P^k_{i,j} = \partial_{xy} D^k_{i,j} \partial_x f^k_{i,j} + \partial_x D^k_{i,j} \partial_{xy} fi,j^k + \partial_{2y} D^k_{i,j} \partial_y f^k_{i,j} + \partial_y f^k_{i,j} \partial_{2y} f^k_{i,j} + \partial_y D^k_{i,j} \left(\partial_{2x} f^k_{i,j} + \partial_{2y} f^k_{i,j} \right) + D^k_{i,j} \left(\partial_{2xy} f^k_{i,j} + \partial_{3y} f^k_{i,j} \right), \tag{64}$$

$$\begin{aligned} \partial_{ty} P^k_{i,j} = {} & \partial_{txy} D^k_{i,j} \partial_x f^k_{i,j} + \partial_{xy} D^k_{i,j} \partial_{tx} fi,j^k + \partial_{tx} D^k_{i,j} \partial_{xy} f^k_{i,j} + \partial_x D^k_{i,j} \partial_{txy} f^k_{i,j} \\ & + \partial_{t2y} D^k_{i,j} \partial_y f^k_{i,j} + \partial_{2y} D^k_{i,j} \partial_{ty} f^k_{i,j} + \partial_{ty} D^k_{i,j} \partial_{2y} f^k_{i,j} + \partial_y D^k_{i,j} \partial_{t2y} f^k_{i,j} \\ & + \partial_{ty} D^k_{i,j} \left(\partial_{2x} f^k_{i,j} + \partial_{2y} f^k_{i,j} \right) + \partial_y D^k_{i,j} \left(\partial_{t2x} f^k_{i,j} + \partial_{t2y} f^k_{i,j} \right) \\ & + \partial_t D^k_{i,j} \left(\partial_{2xy} f^k_{i,j} + \partial_{3y} f^k_{i,j} \right) + D^k_{i,j} \left(\partial_{t2xy} f^k_{i,j} + \partial_{t3y} fi,j^k \right). \end{aligned} \tag{65}$$

In equations (61), (63) and (65), there appear again the terms of $\partial_{tx} f^k_{i,j}$, $\partial_{ty} f^k_{i,j}$, $\partial_{t2x} f^k_{i,j}$, $\partial_{t2y} f^k_{i,j}$, $\partial_{txy} f^k_{i,j}$, $\partial_{t3x} f^k_{i,j}$, $\partial_{t3y} f^k_{i,j}$, $\partial_{t2xy} f^k_{i,j}$, and $\partial_{tx2y} f^k_{i,j}$, to be determined. We can use again the branch equations, equation (55), to the third order for the determination of these terms with use of local values of $\partial_{mxny} f^k_{i,j}$ with $m + n \leq 5$. It should be emphasized here again that when we use the branch equations to the higher order, we can obtain the higher numerical accuracy. This is because by using the branch equations as much as possible, we can avoid the loss of the important information of [Inf.1] on the "relations" to the higher derivatives, which are embedded in the source and branch equations and connecting the local values and their time evolutions.

Using the values at the time of t_k and/or t_{k-1}, we obtain the derivatives of D, for example, $\partial_t D^k_{i,j} = \left(D^k_{i,j} - D^{k-1}_{i,j} \right) / \Delta t$, $\partial_x D^k_{i,j} = \left(D^k_{i+1,j} - D^k_{i-1,j} \right) / 2\Delta x$, $\partial_y D^k_{i,j} = \left(D^k_{i,j+1} - D^k_{i,j-1} \right) / 2\Delta y$, and so on, where Δx $(= x_i - x_{i-1})$ and Δy $(= y_i - y_{i-1})$ are the mesh sizes of x and y directions, respectively. When we use the approximate solutions for $f^{k+1}_{i,j}$, $\partial_x f^{k+1}_{i,j}$ and $\partial_y f^{k+1}_{i,j}$ to the order of $(\Delta t)^2$ in equations (57)–(59), we can determine the values of $f^{k+1}_{i,j}$, $\partial_x f^{k+1}_{i,j}$ and $\partial_y f^{k+1}_{i,j}$ by using equations (60)–(65) and the values of $\partial_{mxny} f^k_{i,j}$ $(m + n \leq 5)$ and D at the time of t_k and t_{k-1}. (If we neglect eighteen terms of $\partial_{mxny} f^k_{i,j}$ $(2 \leq m + n \leq 5)$ in equations (60)–(65) as an approximation, then we can obtain the values of $f^{k+1}_{i,j}$, $\partial_x f^{k+1}_{i,j}$ and $\partial_y f^{k+1}_{i,j}$ without using any additional thought element. In this sense, we are free from using the connection relations of equations (8)–(10) in the fourth thought element [IV] of equation (14). In order to attain higher numerical accuracy, we need the fourth thought element [IV].)

We proceed to the final thought element, [IV], of equation (14). Since we have to determine the values of $f^{k+1}_{i,j}$, $\partial_x f^{k+1}_{i,j}$ and $\partial_y f^{k+1}_{i,j}$ with higher numerical accuracy from those of $f^k_{i,j}$, $\partial_x f^k_{i,j}$ and $\partial_y f^k_{i,j}$, we need the values of $\partial_{mxny} f^k_{i,j}$ $(2 \leq m + n \leq 5)$ included in equations (60)–(65). We therefore use the following interpolation curves up to the terms of $\partial_{mxny} f^k_{i,j}$ with $m + n = 5$ of the Taylor expansion around the point of (x_i, y_j), which represent approximately the rest of the infinite terms of the Taylor expansion:

$$\begin{aligned} F_{i,j}(X,Y) = {} & f^k_{i,j} + \partial_x f^k_{i,j} X + \partial_y f^k_{i,j} Y \\ & + \frac{1}{2} \left(\partial_{2x} f^k_{i,j} X^2 + 2\partial_{xy} f^k_{i,j} XY + \partial_{2y} f^k_{i,j} Y^2 \right) \\ & + \frac{1}{6} \left(\partial_{3x} f^k_{i,j} X^3 + 3\partial_{2xy} f^k_{i,j} X^2 Y + 3\partial_{x2y} f^k_{i,j} XY^2 + \partial_{3y} f^k_{i,j} Y^3 \right) \\ & + \frac{1}{24} \left(\partial_{4x} f^k_{i,j} X^4 + 4\partial_{3xy} f^k_{i,j} X^3 Y + 6\partial_{2x2y} f^k_{i,j} X^2 Y^2 \right. \\ & \qquad \left. + 4\partial_{x3y} f^k_{i,j} XY^3 + \partial_{4y} f^k_{i,j} Y^4 \right) \\ & + \frac{1}{120} \left(\partial_{5x} f^k_{i,j} X^5 + 5\partial_{4xy} f^k_{i,j} X^4 Y + 10\, \partial_{3x2y} f^k_{i,j} X^3 Y^2 \right. \\ & \qquad \left. + 10\, \partial_{2x3y} f^k_{i,j} X^2 Y^3 + 5\partial_{x4y} f^k_{i,j} XY^4 + \partial_{5y} f^k_{i,j} Y^5 \right), \end{aligned} \tag{66}$$

$$\begin{aligned} \partial_x F^k_{i,j}(X,Y) = {} & \partial_x f^k_{i,j} + \partial_{2x} f^k_{i,j} X + \partial_{xy} f^k_{i,j} Y \\ & + \frac{1}{2} \left(\partial_{3x} f^k_{i,j} X^2 + 2\partial_{2xy} f^k_{i,j} XY + \partial_{x2y} f^k_{i,j} Y^2 \right) \end{aligned}$$

$$+\frac{1}{6}\left(\partial_{4x}f^k_{i,j}X^3+3\partial_{3xy}f^k_{i,j}X^2Y+3\partial_{2x2y}f^k_{i,j}XY^2+\partial_{x3y}f^k_{i,j}Y^3\right)$$
$$+\frac{1}{24}\left(\partial_{5x}f^k_{i,j}X^4+4\partial_{4xy}f^k_{i,j}X^3Y+6\partial_{3x2y}f^k_{i,j}X^2Y^2\right.$$
$$\left.+4\,\partial_{2x3y}f^k_{i,j}XY^3+\partial_{x4y}f^k_{i,j}Y^4\right), \tag{67}$$

$$\partial_yF^k_{i,j}(X,Y)=\partial_yf^k_{i,j}+\partial_{xy}f^k_{i,j}X+\partial_{2y}f^k_{i,j}Y$$
$$+\frac{1}{2}\left(\partial_{2xy}f^k_{i,j}X^2+2\partial_{x2y}f^k_{i,j}XY+\partial_{3y}f^k_{i,j}Y^2\right)$$
$$+\frac{1}{6}\left(\partial_{3xy}f^k_{i,j}X^3+3\partial_{2x2y}f^k_{i,j}X^2Y+3\partial_{x3y}f^k_{i,j}XY^2+\partial_{4y}f^k_{i,j}Y^3\right)$$
$$+\frac{1}{24}\left(\partial_{4xy}f^k_{i,j}X^4+4\partial_{3x2y}f^k_{i,j}X^3Y+6\partial_{2x3y}f^k_{i,j}X^2Y^2\right.$$
$$\left.+4\,\partial_{x4y}f^k_{i,j}XY^3+\partial_{5y}f^k_{i,j}Y^4\right), \tag{68}$$

where $X=x-x_i$ and $Y=y-y_j$. In order to fold up the information included in the rest of the infinite terms of the Taylor expansion into the additional eighteen terms of $\partial_{mxny}f^k_{i,j}$ $(2\le m+n\le 5)$ and also to determine their values by $f^k_{i,j}$, $\partial_xf^k_{i,j}$ and $\partial_yf^k_{i,j}$, we use the connection relations of equations (8) and (9) for $f^k_{i,j}$, $\partial_xf^k_{i,j}$ and $\partial_yf^k_{i,j}$ at eight neighboring grid points around the point of (x_i,y_j), as is shown in Figure 3. We have the following twenty-four connection relations at the eight grid points of A $\sim$ H in Figure 3.

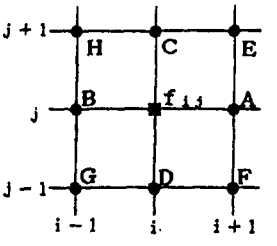

Figure 3. Neighboring grid points used for the connection relations.

$$\text{A}: \quad F^k_{i,j}(\Delta x,0)=f^k_{i+1,j},$$
$$\partial_xF^k_{i,j}(\Delta x,0)=\partial_xf^k_{i+1,j}, \qquad \partial_yF^k_{i,j}(\Delta x,0)=\partial_yf^k_{i+1,j}. \tag{69}$$

$$\text{B}: \quad F^k_{i,j}(0,\Delta y)=f^k_{i,j+1},$$
$$\partial_xF^k_{i,j}(0,\Delta y)=\partial_xf^k_{i,j+1}, \qquad \partial_yF^k_{i,j}(0,\Delta y)=\partial_yf^k_{i,j+1}. \tag{70}$$

$$\text{C}: \quad F^k_{i,j}(\Delta x,\Delta y)=f^k_{i+1,j+1},$$
$$\partial_xF^k_{i,j}(\Delta x,\Delta y)=\partial_xf^k_{i+1,j+1}, \qquad \partial_yF^k_{i,j}(\Delta x,\Delta y)=\partial_yf^k_{i+1,j+1}. \tag{71}$$

$$\text{D}: \quad F^k_{i,j}(-\Delta x,-\Delta y)=f^k_{i-1,j-1},$$
$$\partial_xF^k_{i,j}(-\Delta x,-\Delta y)=\partial_xf^k_{i-1,j-1}, \qquad \partial_yF^k_{i,j}(-\Delta x,-\Delta y)=\partial_yf^k_{i-1,j-1}. \tag{72}$$

$$\text{E}: \quad F^k_{i,j}(-\Delta x,0)=f^k_{i-1,j},$$
$$\partial_xF^k_{i,j}(-\Delta x,0)=\partial_xf^k_{i-1,j}, \qquad \partial_yF^k_{i,j}(-\Delta x,0)=\partial_yf^k_{i-1,j}. \tag{73}$$

$$\text{F}: \quad F^k_{i,j}(\Delta x,-\Delta y)=f^k_{i+1,j-1},$$
$$\partial_xF^k_{i,j}(\Delta x,-\Delta y)=\partial_xf^k_{i+1,j-1}, \qquad \partial_yF^k_{i,j}(\Delta x,-\Delta y)=\partial_yf^k_{i+1,j-1}. \tag{74}$$

$$\text{G}: \quad F^k_{i,j}(0,-\Delta y)=f^k_{i,j-1},$$
$$\partial_xF^k_{i,j}(0,-\Delta y)=\partial_xf^k_{i,j-1}, \qquad \partial_yF^k_{i,j}(0,-\Delta y)=\partial_yf^k_{i,j-1}. \tag{75}$$

$$\mathrm{H}: \quad F^k_{i,j}(-\Delta x, \Delta y) = f^k_{i-1,j+1},$$
$$\partial_x F^k_{i,j}(-\Delta x, \Delta y) = \partial_x f^k_{i-1,j+1}, \qquad \partial_y F^k_{i,j}(-\Delta x, \Delta y) = \partial_y f^k_{i-1,j+1}. \tag{76}$$

These twenty-four connection relations of equations (69)–(76) are reduced to eighteen independent equations after some algebra, and they are solved straightforward in the following forms:

$$\partial_{2x} f^k_{i,j} = \frac{2}{(\Delta x)^2}\left(f^k_{i+1,j} - 2f^k_{i,j} + f^k_{i-1,j}\right) - \frac{1}{2\Delta x}\left(\partial_x f^k_{i+1,j} - \partial_x f^k_{i-1,j}\right), \tag{77}$$

$$\partial_{xy} f^k_{i,j} = \frac{1}{2\Delta x}\left(\partial_y f^k_{i+1,j} - \partial_y f^k_{i-1,j}\right) + \frac{1}{2\Delta y}\left(\partial_x f^k_{i,j+1} - \partial_x f^k_{i,j-1}\right)$$
$$-\frac{1}{4\Delta x \Delta y}\left(f^k_{i+1,j+1} + f^k_{i-1,j-1} - f^k_{i+1,j-1} - f^k_{i-1,j+1}\right), \tag{78}$$

$$\partial_{2y} f^k_{i,j} = \frac{2}{(\Delta y)^2}\left(f^k_{i,j+1} - 2f^k_{i,j} + f^k_{i,j-1}\right) - \frac{1}{2\Delta y}\left(\partial_y f^k_{i,j+1} - \partial_y f^k_{i,j-1}\right), \tag{79}$$

$$\partial_{3x} f^k_{i,j} = \frac{15}{2(\Delta y)^3}\left(f^k_{i+1,j} - f^k_{i-1,j}\right) - \frac{3}{2(\Delta x)^2}\left(\partial_x f^k_{i+1,j} + 8\partial_x f^k_{i,j} + \partial_x f^k_{i-1,j}\right), \tag{80}$$

$$\partial_{2xy} f^k_{i,j} = \frac{1}{2\Delta x \Delta y}\left(-\partial_x f^k_{i+1,j+1} - \partial_x f^k_{i-1,j-1} + \partial_x f^k_{i+1,j} + \partial_x f^k_{i-1,j} - 2\partial_x f^k_{i,j+1}\right.$$
$$\left.+4\partial_x f^k_{i,j} - 2\partial_x f^k_{i,j-1}\right) + \frac{1}{(\Delta x)^2}\left(\partial_y f^k_{i+1,j} - 2\partial_y f^k_{i,j} + \partial_y f^k_{i-1,j}\right)$$
$$+\frac{1}{4(\Delta x)^2 \Delta y}\left(5f^k_{i+1,j+1} - 5f^k_{i-1,j-1} + f^k_{i+1,j-1} - f^k_{i-1,j+1}\right.$$
$$\left.-6f^k_{i+1,j} + 6f^k_{i-1,j} - 4f^k_{i,j+1} + 4f^k_{i,j-1}\right), \tag{81}$$

$$\partial_{x2y} f^k_{i,j} = \frac{1}{2\Delta x \Delta y}\left(-\partial_x f^k_{i+1,j+1} - \partial_y f^k_{i-1,j-1} + \partial_y f^k_{i,j+1} + \partial_y f^k_{i,j-1} - 2\partial_y f^k_{i+1,j}\right.$$
$$\left.+4\partial_y f^k_{i,j} - 2\partial_y f^k_{i-1,j}\right) + \frac{1}{(\Delta y)^2}\left(\partial_x f^k_{i,j+1} - 2\partial_x f^k_{i,j} + \partial_x f^k_{i,j-1}\right)$$
$$+\frac{1}{4\Delta x (\Delta y)^2}\left(5f^k_{i+1,j+1} - 5f^k_{i-1,j-1} - f^k_{i+1,j-1} + f^k_{i-1,j+1}\right.$$
$$\left.-4f^k_{i+1,j} + 4f^k_{i-1,j} - 6f^k_{i,j+1} + 6f^k_{i,j-1}\right), \tag{82}$$

$$\partial_{3y} f^k_{i,j} = \frac{15}{2(\Delta y)^3}\left(f^k_{i,j+1} - f^k_{i,j-1}\right) - \frac{3}{2(\Delta y)^2}\left(\partial_y f^k_{i,j+1} + 8\partial_y f^k_{i,j} + \partial_y f^k_{i,j-1}\right), \tag{83}$$

$$\partial_{4x} f^k_{i,j} = \frac{6}{(\Delta x)^3}\left(\partial_x f^k_{i+1,j} - \partial_x f^k_{i-1,j}\right) - \frac{12}{(\Delta x)^4}\left(f^k_{i+1,j} - 2f^k_{i,j} + f^k_{i-1,j}\right), \tag{84}$$

$$\partial_{3xy} f^k_{i,j} = \frac{3}{2(\Delta x)^3 \Delta y}\left(f^k_{i+1,j+1} + f^k_{i-1,j-1} - f^k_{i+1,j-1} - f^k_{i-1,j+1}\right)$$
$$-\frac{3}{(\Delta x)^2 \Delta y}\left(\partial x f^k_{i,j+1} - \partial x f^k_{i,j-1}\right), \tag{85}$$

$$\partial_{2x2y} f^k_{i,j} = \frac{1}{(\Delta x \Delta y)^2}\left(f^k_{i+1,j+1} + f^k_{i-1,j-1} + f^k_{i+1,j-1} + f^k_{i-1,j+1} + 4f^k_{i,j}\right.$$
$$\left.-2f^k_{i+1,j} - 2f^k_{i-1,j} - 2f^k_{i,j+1} - 2f^k_{i,j-1}\right), \tag{86}$$

$$\partial_{x3y} f^k_{i,j} = \frac{3}{2\Delta x (\Delta y)^3}\left(f^k_{i+1,j+1} + f^k_{i-1,j-1} - f^k_{i+1,j-1} - f^k_{i-1,j+1}\right)$$
$$-\frac{3}{\Delta x (\Delta y)^2}\left(\partial y f^k_{i+1,j} - \partial y f^k_{i-1,j}\right), \tag{87}$$

$$\partial_{4y} f^k_{i,j} = \frac{6}{(\Delta y)^3}\left(\partial_y f^k_{i,j+1} - \partial_y f^k_{i,j-1}\right) - \frac{12}{(\Delta y)^4}\left(f^k_{i,j+1} - 2f^k_{i,j} + f^k_{i,j-1}\right), \tag{88}$$

$$\partial_{5x} f^k_{i,j} = -\frac{90}{2(\Delta x)^5}\left(f^k_{i+1,j} - f^k_{i-1,j}\right) + \frac{30}{2(\Delta x)^4}\left(\partial_x f^k_{i+1,j} + 4\partial_x f^k_{i,j} + \partial_x f^k_{i-1,j}\right), \tag{89}$$

$$\begin{aligned}\partial_{4xy} f^k_{i,j} = {} & \frac{6}{(\Delta x)^3 \Delta y}\left(\partial_x f^k_{i+1,j+1} + \partial_x f^k_{i-1,j-1} + 2\partial_x f^k_{i,j+1} - 4\partial_x f^k_{i,j} + 2\partial_x f^k_{i,j-1}\right. \\ & \left. - \partial_x f^k_{i+1,j} - \partial_x f^k_{i-1,j}\right) - \frac{3}{(\Delta x)^4 \Delta y}\left(5f^k_{i+1,j+1} - 5f^k_{i-1,j-1} + f^k_{i+1,j-1}\right. \\ & \left. - f^k_{i-1,j+1} - 6f^k_{i+1,j} - 6f^k_{i-1,j} - 4f^k_{i,j+1} + 4f^k_{i,j-1}\right),\end{aligned} \tag{90}$$

$$\begin{aligned}\partial_{3x2y} f^k_{i,j} = {} & \frac{3}{(\Delta x)^3(\Delta y)^2}\left(f^k_{i+1,j+1} - f^k_{i-1,j-1} + f^k_{i+1,j-1} - f^k_{i-1,j+1} - 2f^k_{i+1,j} + 2f^k_{i-1,j}\right) \\ & - \frac{6}{(\Delta x)^2(\Delta y)^2}\left(\partial_x f^k_{i,j+1} - 2\partial_x f^k_{i,j} + \partial_x f^k_{i,j-1}\right),\end{aligned} \tag{91}$$

$$\begin{aligned}\partial_{2x3y} f^k_{i,j} = {} & \frac{3}{(\Delta x)^2(\Delta y)^3}\left(f^k_{i+1,j+1} - f^k_{i-1,j-1} - f^k_{i+1,j-1} + f^k_{i-1,j+1} - 2f^k_{i,j+1} + 2f^k_{i,j-1}\right) \\ & - \frac{6}{(\Delta x)^2(\Delta y)^2}\left(\partial_y f^k_{i+1,j} - 2\partial_y f^k_{i,j} + \partial_y f^k_{i-1,j}\right),\end{aligned} \tag{92}$$

$$\begin{aligned}\partial_{x4y} f^k_{i,j} = {} & \frac{6}{\Delta x(\Delta y)^3}\left(\partial_y f^k_{i+1,j+1} + \partial_y f^k_{i-1,j-1} + 2\partial_y f^k_{i+1,j} - 4\partial_y f^k_{i,j} + 2\partial_y f^k_{i-1,j}\right. \\ & \left. - \partial_y f^k_{i,j+1} - \partial_y f^k_{i,j-1}\right) - \frac{3}{\Delta x(\Delta y)^4}\left(5f^k_{i+1,j+1} - 5f^k_{i-1,j-1} - f^k_{i+1,j-1}\right. \\ & \left. + f^k_{i-1,j+1} - 6f^k_{i,j+1} + 6f^k_{i,j-1} - 4f^k_{i+1,j} + 4f^k_{i-1,j}\right),\end{aligned} \tag{93}$$

$$\partial_{5y} f^k_{i,j} = -\frac{90}{(\Delta y)^5}\left(f^k_{i,j+1} - f^k_{i,j-1}\right) + \frac{30}{(\Delta y)^4}\left(\partial_y f^k_{i,j+1} + 4\partial_y f^k_{i,j} + \partial_y f^k_{i,j-1}\right). \tag{94}$$

Using the initial values of $f_{i,j}$, i.e., $f^1_{i,j}$, we obtain the initial values of $\partial_x f_{i,j}$ and $\partial_y f_{i,j}$, by $\partial_x f^1_{i,j} = \left(f^1_{i+1,j} - f^1_{i-1,j}\right)/2\Delta x$ and $\partial_y f^1_{i,j} = \left(f^1_{i,j+1} - f^1_{i,j-1}\right)/2\Delta y$, respectively. We may notice from equations (58)–(94) shown above that we can suppress the losses of [Inf.2] on the functional "values" in $f(t,x,y)$ by folding up the information included in the rest of the infinite terms of the Taylor expansion into the additional four terms of $\partial_{mxny} f^k_{i,j}$ $(2 \le m+n \le 5)$ and by finding all of the values of $\partial_{mxny} f^k_{i,j}$ $(1 \le m+n \le 5)$. If we treat only the values of $f^k_{i,j}$ instead of $\{f^k_{i,j},\ \partial_x f^k_{i,j},\ \partial_y f^k_{i,j}\}$, we do lose the important information of [Inf.2] on the functional "values" shown above, and these losses of [Inf.2] will accumulate numerical error to increase during data processing.

We use the same procedure for the determination of the boundary values of $f^k_{i,j}$, $\partial_x f^k_{i,j}$ and $\partial_y f^k_{i,j}$, as was used for the one-dimensional case shown from equation (39) to equation (50). In the two-dimensional case, we have to deal with two types of boundaries, i.e., the edge boundaries and the corner ones. Using the thought elements [III] and [IV] of equation (14), and reducing some terms from the highest derivative in the Taylor coefficients in order to match the number of unknown Taylor coefficients and that of the connection relations, we can obtain the necessary Taylor coefficients for the determination of the boundary values, in the same way from equation (39) to equation (50). This part of algebra is rather complicated compared with the one-dimensional case, but is straightforward.

Combining all processes for the four thoughts, $\{[\mathrm{I}], [\mathrm{II}], [\mathrm{III}], [\mathrm{IV}]\}$, and for the boundary values, we can find the set of the discrete solutions to the 1st derivatives $\{f^{k+1}_{i,j},\ \partial_x f^{k+1}_{i,j},\ \partial_y f^{k+1}_{i,j}\}$ after one time step from the state of $\{f^k_{i,j},\ \partial_x f^k_{i,j},\ \partial_y f^k_{i,j}\}$.

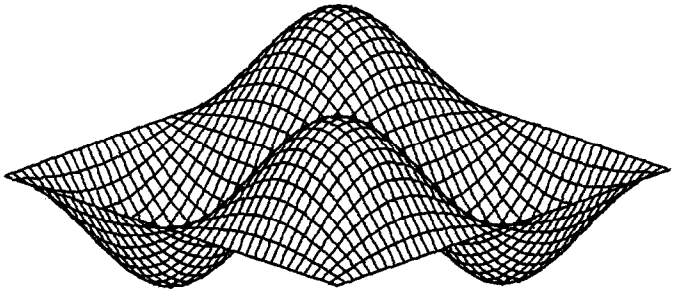

Figure 4. Initial profile of $f^k_{i,j}$ for the two-dimensional diffusion equation.

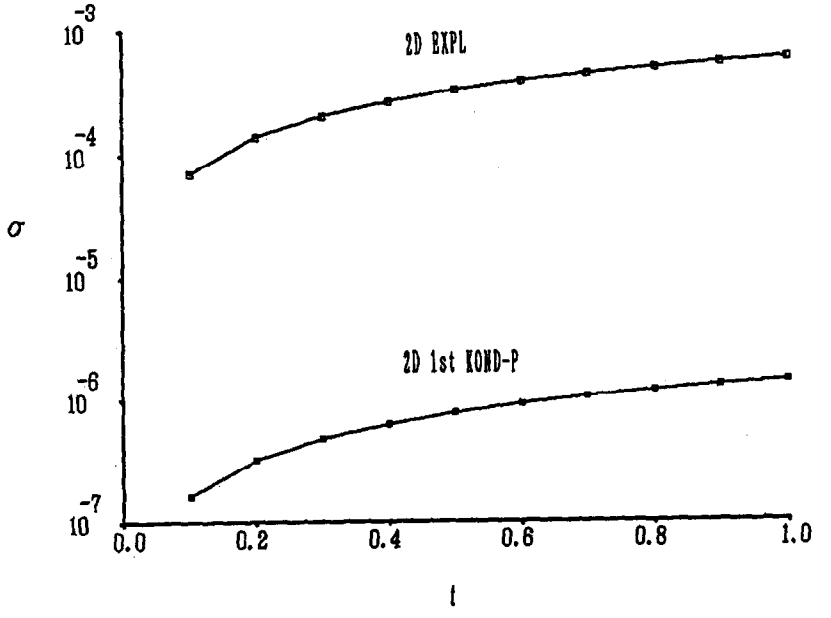

Figure 5. Typical results of computation for the time evolution of σ in the case of $M = 20$ and $D\Delta t/(\Delta x)^2 = 0.1$. Two lines of σ for the 2-D EXPL scheme (the mark □) and the 2-D 1st KOND-P scheme (the mark ■) are shown in a semi-log scale.

We show here some typical numerical results by the 2-D 1st KOND-P scheme shown above, comparing with the results by the two-dimensional explicit scheme denoted by "2-D EXPL" as the reference measure. In order to test the accuracy of the numerical results, we calculate the following case with the analytical solution: the diffusion coefficient $D = 1$, the initial profile of $f(0, x, y) = \sin(2\pi x/\lambda_x)\sin(2\pi y/\lambda_y)$, and the boundary condition (a) of $f^k_{i,j} = 0$ ($i = 1, N$ and $j = 1, N$), where both of the total mesh numbers in the x and the y directions are set to be N, and $\Delta x = \Delta y$, for simplicity. Here λ_x and λ_y are one period length of $f(0, x, y)$ in the x and the y directions, respectively, and we show the case of $\lambda_x = \lambda_y$ in the following example. The analytical solution for this case is written as $f(t, x, y) = \exp\left[-\left(k_x^2 + k_y^2\right)t\right]\sin(k_x x)\sin(k_y y)$, where $k_x = 2\pi/\lambda_x$ and $k_y = 2\pi/\lambda_y$. When we define M as the number of meshes in one period length in the x direction, λ_x and λ_y are now given by $\lambda_x = \lambda_y = M\Delta x$, and $M + 1$ grid points cover one period length. We use the following conventional two-dimensional explicit scheme, obtained from the finite-difference equation: $f^{k+1}_{i,j} = f^k_{i,j} + D\Delta t\left[\left(f^k_{i+1,j} - 2f^k_{i,j} + f^k_{i-1,j}\right)/(\Delta x)^2 + \left(f^k_{i,j+1} - 2f^k_{i,j} + f^k_{i,j-1}\right)/(\Delta y)^2\right]$. The root mean square deviation, σ, from the analytical solution used for the quantitative measurement of the numerical accuracy is now given by

$$\sigma = \left\{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[f^k_{i,j} - f(t_k, x_i, y_j)\right]^2\right\}^{1/2}. \tag{95}$$

The double precision programme is used for the following numerical calculations.

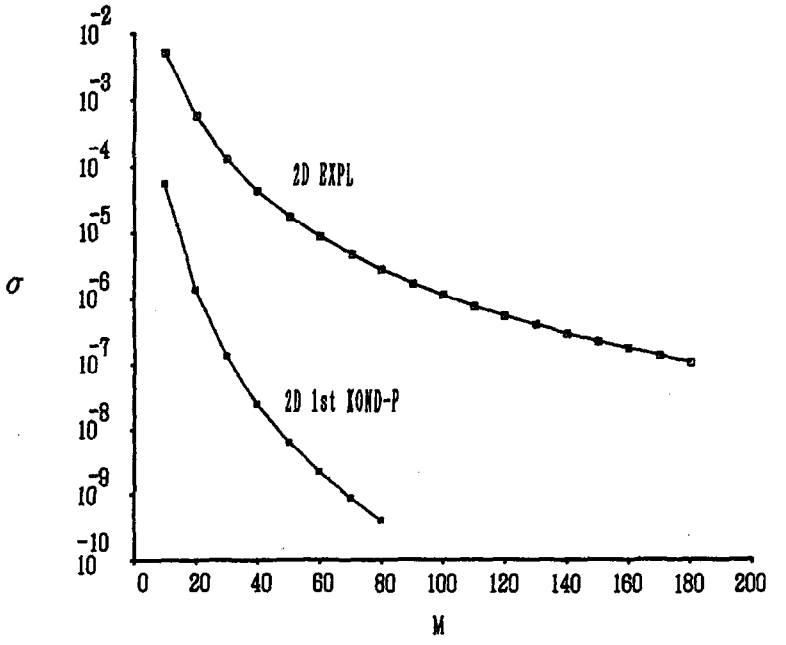

Figure 6. Dependence of numerical error measured by σ on the number of meshes M in one period length in the case of $D\Delta t/(\Delta x)^2 = 0.1$. Two lines of σ at the time of $t = 1.0$ are shown in a semi-log scale for both of the 2-D EXPL scheme (the mark □) and the 2-D 1st KOND-P scheme (the mark ■).

Figure 4 shows the initial profile of $f_{i,j}^k$ used for the calculation of the two-dimensional diffusion equation. Figure 5 shows typical results of computation for the time evolution of σ in the case of $M = 20$ and $D\Delta t/(\Delta x)^2 = 0.1$, where two lines of σ for the 2-D EXPL scheme (the mark □) and the 2-D 1st KOND-P scheme (the mark ■) are shown in a semi-log scale. We recognize from Figure 5 that the error of the 2-D 1st KOND-P scheme measured by σ is less than that of the 2-D EXPL one by over 2 orders in this case. This result is similar to the one-dimensional case in the previous Section 3.1.

Figure 6 shows typical results of computation to see the dependence of σ on the number of meshes M in one period length in the case of $D\Delta t/(\Delta x)^2 = 0.1$, where the values of σ at the time of $t = 1.0$ are shown in a semi-log scale for both of the 2-D EXPL scheme (the mark □) and the 2-D 1st KOND-P scheme (the mark ■). It is recognized from Figure 6 that quite higher improvement rate of accuracy by increasing the number of meshes M can be achieved in the 2-D 1st KOND-P scheme than in the 2-D EXPL one. In the case of $M = 40$, the error of the 2-D 1st KOND-P scheme measured by σ becomes less than that of the 2-D EXPL one by over 3 orders, as is seen in Figure 6.

Figure 7 shows dependence of the CPU time on the number of meshes M for the computation until $t = 1.0$ and with the same parameter of $D\Delta t/(\Delta x)^2 = 0.1$. Here, the CPU time is the computational time on a LUNA 88K SX9100/DT8862 computer system. It is seen from Figure 7 that the CPU times for both the 2-D 1st KOND-P scheme and the 2-D EXPL one increase almost proportional to M^2, as is expected. It is important to compare the CPU time used for the computation by both the schemes to get the same numerical accuracy. We take here the case of $M = 20$ for the 2-D 1st KOND-P scheme as a typical example. In this case, the error by the value of σ is about 10^{-6}, as is seen from Figure 6. In order to attain this accuracy of $\sigma \sim 10^{-6}$, we need the number of meshes of about $M = 90$ for the 2-D EXPL scheme, as is seen from Figure 6. The CPU time for the case of $M = 20$ by the 2-D 1st KOND-P scheme is 130 sec, while the CPU time for the case of $M = 90$ by the 2-D EXPL scheme is 660 sec, as is shown in Figure 7. This result demonstrates that the present 2-D 1st KOND-P scheme yields the same numerical accuracy with $\sigma \sim 10^{-6}$ by using only about 1/5 of the CPU time used by the 2-D EXPL scheme. If we take

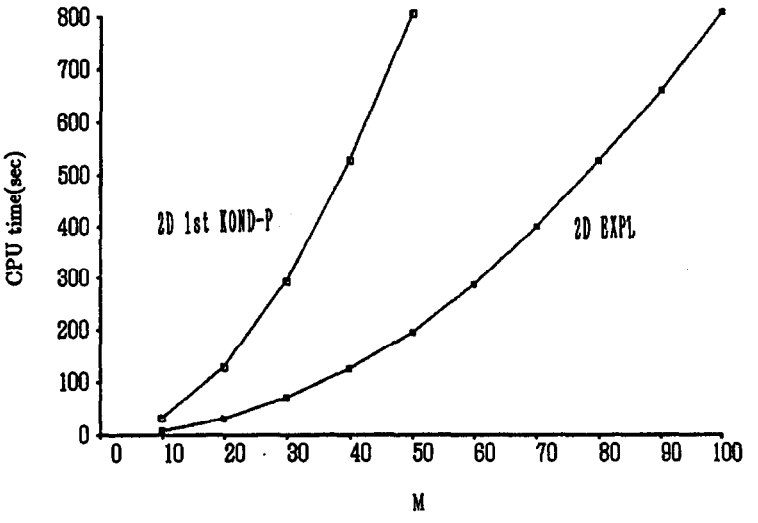

Figure 7. Dependence of the CPU time on the number of meshes M for the computation until $t = 1.0$. $D\Delta t/(\Delta x)^2 = 0.1$.

other case of $M = 30$ for the 2-D 1st KOND-P scheme with the value of $\sigma \sim 10^{-7}$, then we need the number of meshes of about $M = 170$ for the 2-D EXPL scheme, as is seen from Figure 6. In this case with $\sigma \sim 10^{-7}$, the 2-D 1st KOND-P scheme yields the same numerical accuracy by using only about 1/8 of the CPU time used by the 2-D EXPL scheme.

We recognize from the typical numerical results shown above that the KOND algorithm with the four thoughts $\{[\,\mathrm{I}\,], [\mathrm{II}], [\mathrm{III}], [\mathrm{IV}]\}$ gives us the novel numerical scheme of the 2-D 1st KOND-P scheme which yields quite high accuracy and, therefore, effective high reduction of the CPU time to attain the same common numerical accuracy.

4. KOND ALGORITHM FOR HYPERBOLIC EQUATIONS (KOND-H SCHEME)

We apply here the KOND algorithm to the numerical scheme for solving the hyperbolic type equation. In the following Section 4.1, we present the procedure for the development of a scheme to obtain the discrete solutions to the 2nd derivatives, i.e., f_n, $\partial_x f_n$ and $\partial_{2x} f_n$, in order to show the basic processes of the KOND algorithm applied to the one-dimensional hyperbolic equation and the origin of the resultant numerical accuracy. We express here the scheme for the one-dimensional hyperbolic equation to the 2nd derivatives by the KOND algorithm as "1-D 2nd KOND-H scheme." In Section 4.2, we present a 1-D 1st KOND-H scheme, which is the scheme to obtain the discrete solutions to the 1st derivatives for the one-dimensional hyperbolic equation, in order to show some comparisons on the numerical accuracy with the 1-D 2nd KOND-H scheme and the compact CIP scheme which is known to be a less diffusive scheme compared with other conventional schemes. The comparisons of numerical results among the 1-D 2nd KOND-H scheme, the 1-D 1st KOND-H scheme, and the compact CIP scheme are shown in Section 4.3.

4.1. One-Dimensional 2nd Order Case (1-D 2nd KOND-H Scheme)

We treat here a one-dimensional hyperbolic equation, and develop a scheme to obtain the discrete solutions to the second derivatives, i.e., f_n, $\partial_x f_n$, and $\partial_{2x} f_n$. According to the thought element [I] of equation (14), and from equations (4) and (5), the source equation and the 1st and 2nd branch equations are as follows:

$$L_h[f(t,x)] \equiv \partial_t f + u(t,x)\partial_x f, \tag{96}$$

$$L_h[f] = g(t,x), \tag{97}$$

$$L_h[\partial_x f] = \partial_x g - \partial_x u \partial_x f, \tag{98}$$

$$L_h[\partial_{2x} f] = \partial_{2x} g - 2\partial_x u \partial_{2x} f - \partial_{2x} u \partial_x f, \tag{99}$$

where equation (96) is the definition of the operator L_h of the hyperbolic equation, and the notations such as $\partial_t f$, $\partial_x f$, and $\partial_{2x} f$ denote again $\frac{\partial f(t,x)}{\partial t}$, $\frac{\partial f(t,x)}{\partial x}$, $\frac{\partial^2 f(t,x)}{\partial x^2}$, respectively. All of the equations (97)–(99) are the same type of differential equations. By using the branch equations up to the 2nd order, we can suppress the losses of [Inf.1] on the "relations" in the differential equations up to the 2nd order.

According to the thought element [II] of equation (14), we solve equations (97)–(99) locally around a point of (t_0, x_0) as analytically as possible, in order to suppress the losses of [Inf.1] on the "relations" in the differential equations themselves by discretizations. Using the Taylor expansion, we write $u(t,x)$ as

$$u(t,x) = u_0 + \partial_t u_0 \tau + \partial_x u_0 s + \cdots, \tag{100}$$

where $u_0 \equiv u(t_0, x_0)$, $\partial_t u_0 \equiv \partial_t u(t_0, x_0)$, $\partial_x u_0 \equiv \partial_x u(t_0, x_0)$, $\tau \equiv t - t_0$, and $s \equiv x - x_0$. By using equation (100), the source equation and the branch equations, equations (96)–(99), are rewritten as

$$L_{h0}[f(t,x)] \equiv \partial_t f + u_0 \partial_x f, \tag{101}$$

$$L_{h0}[f] = G0(t,x), \tag{102}$$

$$G0(t,x) \equiv g(t,x) - \partial_x f(\partial_t u_0 \tau + \partial_x u_0 s + \cdots), \tag{103}$$

$$L_{h0}[\partial_x f] = G1(t,x), \tag{104}$$

$$G1(t,x) \equiv \partial_x g - \partial_x u \partial_x f - \partial_{2x} f(\partial_t u_0 \tau + \partial_x u_0 s + \cdots), \tag{105}$$

$$L_{h0}[\partial_{2x} f] = G2(t,x), \tag{106}$$

$$G2(t,x) \equiv \partial_{2x} g - 2\partial_x u \partial_{2x} f - \partial_{2x} u \partial_x f - \partial_{3x} f(\partial_t u_0 \tau + \partial_x u_0 s + \cdots), \tag{107}$$

where L_{h0} defined by equation (101) is the operator with a constant value of u_0. We may solve the same type equations of equations (102), (104), and (106) by using the two phases of the Eulerian and the Lagrangean phases, as follows:

⟨Eulerian phase for equation (102) [Non-advection phase]⟩

$$\begin{aligned} \frac{\mathrm{D}f}{\mathrm{D}t} &\equiv \partial_t f + \frac{\mathrm{d}x}{\mathrm{d}t}\partial_x f \\ &\equiv \partial_t f + u_0 \partial_x f = G0(t,x), \end{aligned} \tag{108}$$

where $\frac{\mathrm{d}x}{\mathrm{d}t} = u_0 = \text{const.}$ The function $G0(t,x)$ is expanded as follows along the path of the advection,

$$\begin{aligned} G0(t,x) &= G0_0 + \partial_t G0_0 \tau + \partial_x G0_0 s + \cdots \\ &= G0_0 + \partial_t G0_0 \tau + \partial_x G0_0 u_0 \tau + \cdots, \end{aligned} \tag{109}$$

where $G0_0 \equiv G0(t_0, x_0)$, $\partial_t G0_0 \equiv \partial_t G0(t_0, x_0)$, $\partial_x G0_0 \equiv \partial_x G0(t_0, x_0)$, $\tau \equiv t - t_0$, and $s = u_0 \tau$ is used. Integrating equation (108) with equation (109) with respect to τ along the path during the time interval of Δt, we obtain the solution of the Eulerian phase, $f^*(t,x)$, around the point of (t_0, x_0), as follows:

$$f^*(t,x) = f(t,x) + \Delta f^*, \tag{110}$$

$$\Delta f^* \equiv G0_0 \Delta t + \frac{1}{2}(\partial_t G0_0 + \partial_x G0_0 u_0)(\Delta t)^2 + \cdots. \tag{111}$$

⟨Lagrangean phase for equation (102) [Advection phase]⟩

$$\partial_t f^* + u_0 \partial_x f^* = 0. \tag{112}$$

Since the solution of equation (112) is known to be $f^*(t,x) = F(x-u_0t)$ because of $u_0 = \text{const.}$, the solution of the Lagrangean phase is written as

$$f^*(t+\Delta t, x) = f^*(t, x-u_0\Delta t). \tag{113}$$

Combining the Eulerian and the Lagrangean phases, we finally obtain the high-order local analytic solution for equation (102) around the point of (t_0, x_0) as follows:

$$f(t+\Delta t, x) = f^*(t, x-u_0\Delta t), \tag{114}$$

where $f^*(t,x)$ is given by equations (110) and (111). Using the same process mentioned above, we obtain the high-order local analytic solutions for equations (104) and (106) as follows:

$$\partial_x f(t+\Delta t, x) = \partial_x f^*(t, x-u_0\Delta t), \tag{115}$$

$$\partial_{2x} f(t+\Delta t, x) = \partial_{2x} f^*(t, x-u_0\Delta t), \tag{116}$$

where two functions of $\partial_x f^*(t,x)$ and $\partial_{2x} f^*(t,x)$ are given as

$$\partial_x f^*(t,x) = \partial_x f(t,x) + G1_0\Delta t + \frac{1}{2}(\partial_t G1_0 + \partial_x G1_0 u_0)(\Delta t)^2 + \cdots, \tag{117}$$

$$\partial_{2x} f^*(t,x) = \partial_{2x} f(t,x) + G2_0\Delta t + \frac{1}{2}(\partial_t G2_0 + \partial_x G2_0 u_0)(\Delta t)^2 + \cdots. \tag{118}$$

The first order approximate solution of equations (110), (111), and (114) with respect to Δt yields the ordinary finite-difference equation. This finite-difference equation loses much of the information of [Inf.1] on the "relations" embedded in the given differential equations corresponding to the higher order terms of equation (111) and also embedded in equations (115)–(118) and higher order branch equations. We use here the second order approximate analytic solutions with respect to Δt in equations (111), (117), and (118), for simplicity.

We now proceed to the third thought element, [III], of equation (14). In order to determine the values of the solutions at the next time step from equations (114)–(116), we need the interpolation curves for the functions of $f^*(t,x)$, $\partial_x f^*(t,x)$ and $\partial_{2x} f^*(t,x)$. The interpolation curves with use of the Taylor expansion, equations (11), (12), (13), ..., are determined by the discrete solutions of f_n^*, $\partial_x f_n^*$, $\partial_{2x} f_n^*$, and so on. We represent here again the discrete solutions at the time of $k\Delta t$ on the grid point x_n as f_n^k, $\partial_x f_n^k$ and $\partial_{2x} f_n^k$. Using the interpolation curves of $F_n^{k*}(s)$, $\partial_x F_n^{k*}(s)$ and $\partial_{2x} F_n^{k*}(s)$, respectively, for $f^{k*}(x)$, $\partial_x f^{k*}(x)$ and $\partial_{2x} f^{k*}(x)$, we obtain the discrete solutions at the time of $(k+1)\Delta t$ from equations (114)–(116) as follows:

$$f_n^{k+1} = F_n^{k*}(r), \tag{119}$$

$$\partial_x f_n^{k+1} = \partial_x F_n^{k*}(r), \tag{120}$$

$$\partial_{2x} f_n^{k+1} = \partial_{2x} F_n^{k*}(r), \tag{121}$$

where $r = -u_n^k\Delta t$, and $u_n^k = u(k\Delta t, x_n)$. [If we use only the terms of f_n^*, $\partial_x f_n^*$, and $\partial_{2x} f_n^*$ for the interpolation curves of equations (11)–(13) by neglecting the terms beyond $\partial_{2x} f_n^*$, we can still obtain f_n^{k+1}, $\partial_x f_n^{k+1}$ and $\partial_{2x} f_n^{k+1}$ from f_n^k, $\partial_x f_n^k$ and $\partial_{2x} f_n^k$ by using equations (119)–(121) without any additional thought element. In this sense, we are free from using the connection relations of equations (8), (9) and (10) for the determination of the discrete solutions.]

We proceed to the final thought element, [IV], of equation (14). We use here three connection relations to the 2nd partial derivative, equations (8)–(10), and therefore, we introduce three additional terms of the Taylor expansion beyond the term of $\partial_{2x} f_n^{k*}$. Using equations (11)–(13) for the interpolation curves and equations (8)–(10) for the three connection relations at the left neighboring grid point x_{n-1} for the case of $u_n^k > 0$, we write the interpolation curves and the

connection relations as follows:

$$F_n^{k*}(s) = f_n^{k*} + \partial_x f_n^{k*} s + \frac{1}{2}\partial_{2x} f_n^{k*} s^2 + \frac{1}{6}\partial_{3x} f_n^{k*} s^3 + \frac{1}{24}\partial_{4x} f_n^{k*} s^4 + \frac{1}{120}\partial_{5x} f_n^{k*} s^5, \quad (122)$$

$$\partial_x F_n^{k*}(s) = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} s + \frac{1}{2}\partial_{3x} f_n^{k*} s^2 + \frac{1}{6}\partial_{4x} f_n^{k*} s^3 + \frac{1}{24}\partial_{5x} f_n^{k*} s^4, \quad (123)$$

$$\partial_{2x} F_n^{k*}(s) = \partial_{2x} f_n^{k*} + \partial_{3x} f_n^{k*} s + \frac{1}{2}\partial_{4x} f_n^{k*} s^2 + \frac{1}{6}\partial_{5x} f_n^{k*} s^3, \quad (124)$$

$$F_n^{k*}(-h) = f_{n-1}^{k*}, \quad (125)$$

$$\partial_x F_n^{k*}(-h) = \partial_x f_{n-1}^{k*}, \quad (126)$$

$$\partial_{2x} F_n^{k*}(-h) = \partial_{2x} f_{n-1}^{k*}, \quad (127)$$

where $s = x - x_n$, and $\partial_{3x} f_n^{k*}$, $\partial_{4x} f_n^{k*}$ and $\partial_{5x} f_n^{k*}$ are the Taylor coefficients of the additional terms. Using the three connection relations of equations (125)–(127) at the left neighboring grid point x_{n-1} for the case of $u_n^k > 0$, we obtain the three additional Taylor coefficients, $\partial_{3x} f_n^{k*}$, $\partial_{4x} f_n^{k*}$ and $\partial_{5x} f_n^{k*}$, which carry the semiglobal information for the uncovered regions between the neighboring grid points, as follows:

$$\partial_{3x} f_n^{k*} = \frac{60}{h^3}(f_n^{k*} - f_{n-1}^{k*}) - \frac{12}{h^2}(3\partial_x f_n^{k*} + 2\partial_x f_{n-1}^{k*}) + \frac{3}{h}(3\partial_{2x} f_n^{k*} - \partial_{2x} f_{n-1}^{k*}), \quad (128)$$

$$\partial_{4x} f_n^{k*} = \frac{360}{h^4}(f_n^{k*} - f_{n-1}^{k*}) - \frac{24}{h^3}(8\partial_x f_n^{k*} + 7\partial_x f_{n-1}^{k*}) + \frac{12}{h^2}(3\partial_{2x} f_n^{k*} - 2\partial_{2x} f_{n-1}^{k*}), \quad (129)$$

$$\partial_{5x} f_n^{k*} = \frac{720}{h^5}(f_n^{k*} - f_{n-1}^{k*}) - \frac{360}{h^4}(\partial_x f_n^{k*} + \partial_x f_{n-1}^{k*}) + \frac{60}{h^3}(\partial_{2x} f_n^{k*} - \partial_{2x} f_{n-1}^{k*}), \quad (130)$$

where h $(= x_n - x_{n-1})$ is the mesh size. From comparison between equations (11)–(13) and equations (122)–(124), we see that we have folded up the rest of the infinite terms beyond the term of $\partial_{2x} f_n^{k*}$ in the Taylor expansions approximately into the three additional terms of $\partial_{3x} f_n^{k*}$, $\partial_{4x} f_n^{k*}$ and $\partial_{5x} f_n^{k*}$ by using the connection relations, as was discussed in Section 2.

We may notice from equations (115)–(130) shown above that we can suppress the losses of [Inf.2] on the functional "values" in $f(t,x)$ by finding the values of $\partial_{mx} f_n^{k*}$ $(m = 1,2,3,4,5)$ and folding up the information included in the rest of the infinite terms of the Taylor expansion into the additional three terms of $\partial_{mx} f_n^{k*}$ $(m = 3,4,5)$. If we treat only the values of f_n^{k*} instead of $\{f_n^{k*}, \partial_x f_n^{k*}, \partial_{2x} f_n^{k*}\}$, we do lose the important information of [Inf.2] on the functional "values" shown above, and these losses of [Inf.2] will accumulate numerical error to increase during data processing.

Combining all above processes for the four elements of thoughts, [I], [II], [III], and [IV], we find the set of the discrete solutions up to the 2nd derivatives $\{f_n^{k+1}, \partial_x f_n^{k+1}, \partial_{2x} f_n^{k+1}\}$ after one time step Δt from the state at the time of $k\Delta t$ as follows:

$$f_n^{k+1} = f_n^{k*} + \partial_x f_n^{k*} r + \frac{1}{2}\partial_{2x} f_n^{k*} r^2 + \frac{1}{6}\partial_{3x} f_n^{k*} r^3 + \frac{1}{24}\partial_{4x} f_n^{k*} r^4 + \frac{1}{120}\partial_{5x} f_n^{k*} r^5, \quad (131)$$

$$\partial_x f_n^{k+1} = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} r + \frac{1}{2}\partial_{3x} f_n^{k*} r^2 + \frac{1}{6}\partial_{4x} f_n^{k*} r^3 + \frac{1}{24}\partial_{5x} f_n^{k*} r^4, \quad (132)$$

$$\partial_{2x} f_n^{k+1} = \partial_{2x} f_n^{k*} + \partial_{3x} f_n^{k*} r + \frac{1}{2}\partial_{4x} f_n^{k*} r^2 + \frac{1}{6}\partial_{5x} f_n^{k*} r^3, \quad (133)$$

where $r = -u_n^k \Delta t$, and $\partial_{3x} f_n^{k*}$, $\partial_{4x} f_n^{k*}$ and $\partial_{5x} f_n^{k*}$ are given by equations (128)–(130) with the condition that h and the subscript $n-1$ are replaced, respectively, by $-h$ and $n+1$ for the case of $u_n^k < 0$. The values of f_n^{k*}, $\partial_x f_n^{k*}$, and $\partial_{2x} f_n^{k*}$ are given as follows to the order of $(\Delta t)^2$, by using equations (110), (111), (117) and (118) with the definitions of equations (103), (105) and (107):

$$f_n^{k*} = f_n^k + g_n^k \Delta t + \frac{1}{2}\left[\partial_t g_n^k + \partial_x g_n^k u_n^k - \left(\partial_t u_n^k + \partial_x u_n^k u_n^k\right)\partial_x f_n^k\right](\Delta t)^2, \quad (134)$$

$$\partial_x f_n^{k*} = \partial_x f_n^k + \left(\partial_x g_n^k - \partial_x u_n^k \partial_x f_n^k\right) \Delta t + \frac{1}{2}\Big[\partial_{tx} g_n^k - \partial_{tx} u_n^k \partial_x f_n^k - \partial_x u_n^k \partial_{tx} f_n^k - \partial_t u_n^k \partial_{2x} f_n^k + \left(\partial_{2x} g_n^k - \partial_{2x} u_n^k \partial_x f_n^k - 2\partial_x u_n^k \partial_{2x} f_n^k\right) u_n^k\Big](\Delta t)^2, \tag{135}$$

$$\partial_{2x} f_n^{k*} = \partial_{2x} f_n^k + \left(\partial_{2x} g_n^k - 2\partial_x u_n^k \partial_{2x} f_n^k - \partial_{2x} u_n^k \partial_x f_n^k\right) \Delta t + \frac{1}{2}\Big[\partial_{t2x} g_n^k - 2\partial_{tx} u_n^k \partial_{2x} f_n^k - 2\partial_x u_n^k \partial_{t2x} f_n^k - \partial_{t2x} u_n^k \partial_x f_n^k - \partial_{2x} u_n^k \partial_{tx} f_n^k - \partial_t u_n^k \partial_{3x} f_n^k + \left(\partial_{3x} g_n^k - 3\partial_{2x} u_n^k \partial_{2x} f_n^k - 3\partial_x u_n^k \partial_{3x} f_n^k - \partial_{3x} u_n^k \partial_x f_n^k\right) u_n^k\Big](\Delta t)^2. \tag{136}$$

Here, the right hand side of equations (134)–(136) are all given by the values at the time of k and/or $(k-1)$, for example, $\partial_x g_n^k = (g_{n+1}^k - g_{n-1}^k)/2h$, $\partial_t g_n^k = (g_n^k - g_n^{k-1})/\Delta t$, and so on.

We may notice from equations (131)–(136) shown above that we can suppress the loss of [Inf.1] on the "relations" to the second order derivatives, which are embedded in the source and the branch equations to the second order and connecting the local values and their time evolutions during data processing.

We may find from the detailed application of the KOND algorithm to the hyperbolic type equation, shown above, that in order to attain higher numerical accuracy, we need to suppress both of the losses of [Inf.1] on the "relations" in the differential equations and of [Inf.2] on the functional "values" in $f(t,x)$ and its derivatives $\partial_{mx} f(t,x)$, to the higher order as possible, during data processing.

4.2. One-Dimensional 1st Order Case (1-D 1st KOND-H Scheme)

In order to compare numerical accuracy and its origin with the 1-D 2nd KOND-H scheme shown in the previous Section 4.1, we develop here a 1-D 1st KOND-H scheme to obtain the discrete solutions of the one-dimensional hyperbolic equation to the first derivatives, i.e., f_n and $\partial_x f_n$. According to the thought element [I] of equation (14), and from equations (4) and (5), the source equation and the branch equation are as follows:

$$L_h[f(t,x)] \equiv \partial_t f + u(t,x)\partial_x f, \tag{137}$$

$$L_h[f] = g(t,x), \tag{138}$$

$$L_h[\partial_x f] = \partial_x g - \partial_x u \partial_x f, \tag{139}$$

where these equations are the same with equations (96)–(98).

According to the thought element [II] of equation (14), we solve equations (138)–(139) locally around a point of (t_0, x_0) as analytically as possible, in order to suppress the losses of [Inf.1] on the "relations" in the differential equations themselves by discretizations. The solutions have been solved in the previous Section 4.1 and are written as

$$f(t+\Delta t, x) = f^*(t, x - u_0 \Delta t), \tag{140}$$

$$\partial_x f(t+\Delta t, x) = \partial_x f^*(t, x - u_0 \Delta t), \tag{141}$$

where two functions of $f^*(t,x)$ and $\partial_x f^*(t,x)$ are given as

$$f^*(t,x) = f(t,x) + G0_0 \Delta t + \frac{1}{2}\left(\partial_t G0_0 + \partial_x G0_0 u_0\right)(\Delta t)^2 + \cdots, \tag{142}$$

$$\partial_x f^*(t,x) = \partial_x f(t,x) + G1_0 \Delta t + \frac{1}{2}\left(\partial_t G1_0 + \partial_x G1_0 u_0\right)(\Delta t)^2 + \cdots. \tag{143}$$

We proceed to the third thought element, [III], of equation (14). In order to determine the values of the solutions at the next time step from equations (140)–(143), we use the interpolation curves of $F_n^{k*}(s)$ and $\partial_x F_n^{k*}(s)$, respectively, for $f^{k*}(x)$ and $\partial_x f^{k*}(x)$, and obtain the discrete solutions at the time of $(k+1)\Delta t$ from equations (140) and (141) as follows:

$$f_n^{k+1} = F_n^{k*}(r), \tag{144}$$

$$\partial_x f_n^{k+1} = \partial_x F_n^{k*}(r), \tag{145}$$

where $r = -u_n^k \Delta t$, and $u_n^k = u(k\Delta t, x_n)$.

We proceed to the final thought element, [IV], of equation (14). We consider here the following two cases of the connection relations:

(α) two connection relations to the 1st partial derivatives, and

(β) three connection relations to the 1st partial derivatives.

In the case of (α), we introduce two additional terms in the Taylor expansion beyond the term of $\partial_x f_n^{k*}$. Using equations (11) and (12) for the interpolation curves and equations (8) and (9) for the two connection relations at the left neighboring grid point x_{n-1} for the case of $u_n^k > 0$, we write the interpolation curves and the connection relations as follows:

$$F_n^{k*}(s) = f_n^{k*} + \partial_x f_n^{k*} s + \frac{1}{2}\partial_{2x} f_n^{k*} s^2 + \frac{1}{6}\partial_{3x} f_n^{k*} s^3, \tag{146}$$

$$\partial_x F_n^{k*}(s) = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} s + \frac{1}{2}\partial_{3x} f_n^{k*} s^2, \tag{147}$$

$$F_n^{k*}(-h) = f_{n-1}^{k*}, \tag{148}$$

$$\partial_x F_n^{k*}(-h) = \partial_x f_{n-1}^{k*}, \tag{149}$$

where $s = x - x_n$, and $\partial_{2x} f_n^{k*}$ and $\partial_{3x} f_n^{k*}$ are the Taylor coefficients of the additional terms. Using the two connection relations of equations (148) and (149) at the left neighboring grid point x_{n-1} for the case of $u_n^k > 0$, we obtain the two additional Taylor coefficients, $\partial_{2x} f_n^{k*}$ and $\partial_{3x} f_n^{k*}$, which carry the semiglobal information for the uncovered regions between the neighboring grid points, as follows:

$$\partial_{2x} f_{n\alpha}^{k*} = -\frac{6}{h^2}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{2}{h}\left(2\partial_x f_n^{k*} + \partial_x f_{n-1}^{k*}\right), \tag{150}$$

$$\partial_{3x} f_{n\alpha}^{k*} = -\frac{12}{h^3}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{6}{h^2}\left(\partial_x f_n^{k*} + \partial_x f_{n-1}^{k*}\right), \tag{151}$$

where $h\ (= x_n - x_{n-1})$ is the mesh size, and the subscript α denotes the case of (α). We express here the numerical scheme by the case of (α) as "the 1-D 1st KOND-Hα scheme."

In the case of (β), we introduce three additional terms in the Taylor expansion beyond the term of $\partial_x f_n^{k*}$, and we write the interpolation curves and three connection relations for the case of $u_n^k > 0$ as follows:

$$F_n^{k*}(s) = f_n^{k*} + \partial_x f_n^{k*} s + \frac{1}{2}\partial_{2x} f_n^{k*} s^2 + \frac{1}{6}\partial_{3x} f_n^{k*} s^3 + \frac{1}{24}\partial_{4x} f_n^{k*} s^4, \tag{152}$$

$$\partial_x F_n^{k*}(s) = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} s + \frac{1}{2}\partial_{3x} f_n^{k*} s^2 + \frac{1}{6}\partial_{4x} f_n^{k*} s^3, \tag{153}$$

$$F_n^{k*}(-h) = f_{n-1}^{k*}, \tag{154}$$

$$\partial_x F_n^{k*}(-h) = \partial_x f_{n-1}^{k*}, \tag{155}$$

$$\partial_x F_n^{k*}(h) = \partial_x f_{n+1}^{k*}, \tag{156}$$

where $\partial_{2x} f_n^{k*}$, $\partial_{3x} f_n^{k*}$ and $\partial_{4x} f_n^{k*}$ are now the Taylor coefficients of the additional terms. Here, after some trial computations we have used the connection relation of the first derivative at

the right neighboring grid point for the third connection relation, equation (156), in order to suppress the loss of [Inf.2] on the functional "values" of the first derivative $\partial_x f(t,x)$. Using the three connection relations equations (154)–(156) at the left and the right neighboring grid points for the case of $u_n^k > 0$, we obtain the three additional Taylor coefficients, $\partial_{2x} f_n^{k*}$, $\partial_{3x} f_n^{k*}$, and $\partial_{4x} f_n^{k*}$, as follows:

$$\partial_{2x} f_n^{k*} = -\frac{4}{h^2}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{1}{6h}\left(\partial_x f_{n+1}^{k*} + 16\partial_x f_n^{k*} + 7\partial_x f_{n-1}^{k*}\right), \tag{157}$$

$$\partial_{3x} f_n^{k*} = \frac{1}{h^2}\left(\partial_x f_{n+1}^{k*} - 2\partial_x f_n^{k*} + \partial_x f_{n-1}^{k*}\right), \tag{158}$$

$$\partial_{4x} f_n^{k*} = \frac{24}{h^4}(f_n^{k*} - f_{n-1}^{k*}) + \frac{6}{h^3}\left(\partial_x f_{n+1}^{k*} + 8\partial_x f_n^{k*} + 3\partial_x f_{n-1}^{k*}\right). \tag{159}$$

When we use the three additional Taylor coefficients given by equations (157)–(159), we find that there appear some noises in the resultant numerical solutions, while the numerical diffusion in the computation of the first derivative $\partial_x f(t,x)$ is suppressed effectively, compared with the case of (α). Therefore, in order to suppress both the numerical diffusion and the numerical noises, we have combined the additional Taylor coefficients by equations (157)–(159) with those by equations (150) and (151), by using numerical weighting factors. After some trial computations, we have found that the following Taylor coefficients, obtained by a combination of 0.6 times equations (157)–(159) and 0.4 times equations (150) and (151), yield fairly good suppression of both the numerical diffusion and the numerical noises,

$$\partial_{2x} f_{n\beta}^{k*} = -\frac{24}{5h^2}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{1}{10h}\left(\partial_x f_{n+1}^{k*} + 32\partial_x f_n^{k*} + 15\partial_x f_{n-1}^{k*}\right), \tag{160}$$

$$\partial_{3x} f_{n\beta}^{k*} = -\frac{24}{5h^3}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{3}{5h^2}\left(\partial_x f_{n+1}^{k*} + 2\partial_x f_n^{k*} + 5\partial_x f_{n-1}^{k*}\right), \tag{161}$$

$$\partial_{4x} f_{n\beta}^{k*} = \frac{14.4}{h^4}\left(f_n^{k*} - f_{n-1}^{k*}\right) + \frac{3.6}{h^3}\left(\partial_x f_{n+1}^{k*} + 8\partial_x f_n^{k*} + 3\partial_x f_{n-1}^{k*}\right). \tag{162}$$

We use here the subscript β for the combination mentioned above, and express the numerical scheme obtained by equations (160)–(162) as "the 1-D 1st KOND-Hβ scheme" hereafter.

Combining all above processes for the four elements of thoughts, [I], [II], [III], and [IV], we find the set of the discrete solutions to the 1st derivatives $\left\{f_n^{k+1}, \partial_x f_n^{k+1}\right\}$ after one time step Δt from the state at the time of $k\Delta t$ as follows.

In the case of (α) [the 1-D 1st KOND-Hα scheme]:

$$f_n^{k+1} = f_n^{k*} + \partial_x f_n^{k*} r + \frac{1}{2}\partial_{2x} f_n^{k*} r^2 + \frac{1}{6}\partial_{3x} f_n^{k*} r^3, \tag{163}$$

$$\partial_x f_n^{k+1} = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} r + \frac{1}{2}\partial_{3x} f_n^{k*} r^2, \tag{164}$$

where $r = -u_n^k \Delta t$, and $\partial_{2x} f_n^{k*}$ and $\partial_{3x} f_n^{k*}$ are given by equations (150) and (151) with the condition that h and the subscript $n-1$ are replaced, respectively, by $-h$ and $n+1$ for the case of $u_n^k < 0$.

In the case of (β) [the 1-D 1st KOND-Hβ scheme]:

$$f_n^{k+1} = f_n^{k*} + \partial_x f_n^{k*} r + \frac{1}{2}\partial_{2x} f_n^{k*} r^2 + \frac{1}{6}\partial_{3x} f_n^{k*} r^3 + \frac{1}{24}\partial_{4x} f_n^{k*} r^4, \tag{165}$$

$$\partial_x f_n^{k+1} = \partial_x f_n^{k*} + \partial_{2x} f_n^{k*} r + \frac{1}{2}\partial_{3x} f_n^{k*} r^2 + \frac{1}{6}\partial_{4x} f_n^{k*} r^3, \tag{166}$$

where $r = -u_n^k \Delta t$, and $\partial_{2x} f_n^{k*}$, $\partial_{3x} f_n^{k*}$ and $\partial_{4x} f_n^{k*}$ are given by equations (160)–(162) with the condition that h and the subscript $n-1$ are replaced, respectively, by $-h$ and $n+1$ for the case of $u_n^k < 0$.

In both the 1-D 1st KOND-Hα and the 1-D 1st KOND-Hβ schemes, the values of f_n^{k*} and $\partial_x f_n^{k*}$ are given as follows to the order of $(\Delta t)^2$, by using equations (142) and (143) with the definitions of equations (103) and (105):

$$f_n^{k*} = f_n^k + g_n^k \Delta t + \frac{1}{2}\left[\partial_t g_n^k + \partial_x g_n^k u_n^k - \left(\partial_t u_n^k + \partial_x u_n^k u_n^k\right)\partial_x f_n^k\right](\Delta t)^2, \tag{167}$$

$$\begin{aligned}\partial_x f_n^{k*} = {} & \partial_x f_n^k + \left(\partial_x g_n^k - \partial_x u_n^k \partial_x f_n^k\right)\Delta t \\ & + \frac{1}{2}\Big[\partial_{tx} g_n^k - \partial_{tx} u_n^k \partial_x f_n^k - \partial_x u_n^k \partial_{tx} f_n^k - \partial_t u_n^k \partial_{2x} f_n^k \\ & + \left(\partial_{2x} g_n^k - \partial_{2x} u_n^k \partial_x f_n^k - 2\partial_x u_n^k \partial_{2x} f_n^k\right) u_n^k\Big](\Delta t)^2.\end{aligned} \tag{168}$$

4.3. Comparison of Numerical Results

We show here some typical numerical results by the 1-D 2nd KOND-H scheme, the 1-D 1st KOND-Hα scheme, and the 1-D 1st KOND-Hβ scheme shown in the previous subsections, in order to compare their numerical accuracy and the origin of numerical error. We also compare the numerical results by the three KOND-H schemes with those by the compact CIP scheme [15–18] which is known to be less diffusive compared with other conventional schemes such as FCT [6], QUICKEST [7], TVD [10,11], and PPM [12,13].

Figure 8 shows typical results of computation for the linear wave propagation of $f(t,x)$ for the case of a triangular wave after 1000 time steps, where (a), (b), (c), and (d) are the results, respectively, by the 1-D 2nd KOND-H scheme, the 1-D 1st KOND-Hα scheme, the 1-D 1st KOND-Hβ scheme, and the compact CIP scheme without interpolation check [16–18]. In the figure, the numerical data of f_n and $\partial_x f_n$ are shown by the ♦ marks together with the analytical profile of $f(t,x)$ by the solid lines. The raw numerical data of the peak point, f_p, corresponding to the analytical value of 1.0 at the peak point of the triangular wave are also shown for the four schemes in Figure 8, together with the relative error at the peak point, defined by $\Delta = (1.0 - f_p)/100$, and the numerical error measured by σ. We see from Figure 8a that the value of Δ is only about 2.2% and the step function-like profile of $\partial_x f_n$ is well-realized until 1000 time steps in the 1-D 2nd KOND-H scheme. We recognize from the comparison between the 1-D 2nd KOND-H scheme of Figure 8a and the 1-D 1st KOND-Hα scheme of Figure 8b that the numerical diffusion increases fairly large, as is seen clearly by the profiles of $\partial_x f_n$ and by both the increments of the Δ value from 2.2% to 8.4% and the σ value from 0.0025 to 0.0077. This result comes from the structural difference between the 1-D 2nd KOND-H and the 1-D 1st KOND-Hα schemes that affects on both the losses of [Inf.1] on the "relations" and of [Inf.2] on the functional "values" up to the 1st derivative $\partial_x f$ or to the 2nd derivative $\partial_{2x} f$, as is seen from the derivations of both the schemes in Sections 4.1 and 4.2. The numerical accuracy of the 1-D 1st KOND-Hβ scheme is improved, compared with the 1-D 1st KOND-Hα scheme, as is seen clearly by the profiles of $\partial_x f_n$ and by both the decrements of the Δ value from 8.4% to 6.9% and the σ value from 0.0077 to 0.0062 (Figures 8b and 8c). This result comes from the suppression of the loss of [Inf.2] on the "values" by using more elements of the connection relations in the 1-D 1st KOND-Hβ scheme than in the 1-D 1st KOND-Hα one, as is seen from the derivations of both the schemes in Section 4.2.

We recognize from comparison among the three figures of Figures 8a, 8c, and 8d that the two of the 1-D 2nd KOND-H and the 1-D 1st KOND-Hβ schemes yield even less diffusive error, compared with the compact CIP scheme [15–18] which is known to be less diffusive compared with other conventional schemes such as FCT [6], QUICKEST [7], TVD [10,11], and PPM [12,13]. The 1-D 1st KOND-Hα scheme is seen to yield almost the same diffusive error with the compact CIP scheme (Figures 8b and 8d). It will be demonstrated later, however, that the 1-D 1st KOND-Hα scheme yields much less diffusive error compared with the compact CIP scheme for the case of the computations of the nonlinear wave propagation such as the soliton wave.

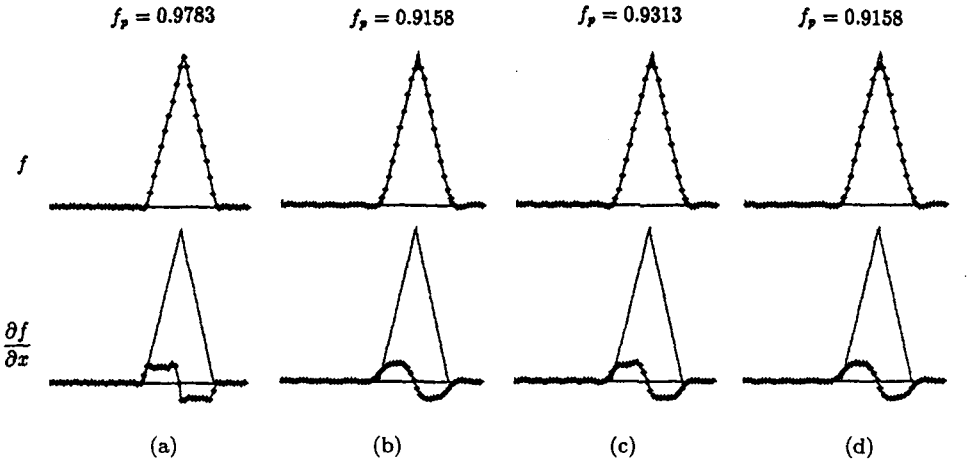

Figure 8. Typical results of computation for the linear wave propagation for the case of a triangular wave after 1000 time steps, $k(= c\Delta t/\Delta x)$ being 0.2. (a) 1-D 2nd KOND-H scheme, $\Delta = 2.2\%$, $\sigma = 0.0025$. (b) 1-D 1st KOND-Hα scheme, $\Delta = 8.4\%$, $\sigma = 0.0077$. (c) 1-D 1st KOND-Hβ scheme, $\Delta = 6.9\%$, $\sigma = 0.0062$. (d) Compact CIP scheme without interpolation check, $\Delta = 8.4\%$, $\sigma = 0.0077$. Numerical data of f_n and $\partial_x f_n$ are shown by the ♦ marks together with the analytical profile of $f(t, x)$ by the solid lines. f_p denotes raw numerical data of the peak point, corresponding to the analytical value of 1.0 at the peak point of the triangular wave. Δ denotes the relative error at the peak point, defined by $\Delta = (1.0 - f_p)/100$.

Figure 9 shows typical results of computation for the linear wave propagation of $f(t, x)$ for the case of a square wave after 1000 time steps, where (a), (b), (c), and (d) are the results, respectively, by the 1-D 2nd KOND-H scheme, the 1-D 1st KOND-Hα scheme, the 1-D 1st KOND-Hβ scheme, and the compact CIP scheme without interpolation check. In the figure, the numerical data of f_n and $\partial_x f_n$ are shown again by the ♦ marks together with the analytical profile of $f(t, x)$ by the solid lines. We see from Figure 9a that the transition region of the square wave is expressed only by two data points and the delta function-like profile of $\partial_x f_n$ is nearly realized until 1000 time steps in the 1-D 2nd KOND-H scheme. We recognize from the comparison between the 1-D 2nd KOND-H scheme of Figure 9a and the 1-D 1st KOND-Hα scheme of Figure 9b that the numerical diffusion increases fairly large, as is seen clearly by both the profiles of f_n and $\partial_x f_n$ and by the increment of the σ value from 0.027 to 0.052. The numerical accuracy of the 1-D 1st KOND-Hβ scheme is improved, compared with the 1-D 1st KOND-Hα scheme, as is seen by the amplitude of $\partial_x f_n$ profile and by the decrement of the σ value from 0.052 to 0.048 (Figures 9b and 9c). The 1-D 1st KOND-Hα scheme yields almost the same diffusive error with the compact CIP scheme (Figures 9b and 9d). These features of the numerical results for the square wave are the same with those for the triangular wave, and the discussions for the results of the triangular wave mentioned above are also applicable to these results of the square wave.

As a typical example of the nonlinear wave propagation, we next show other results of computation for the soliton wave described by the Korteweg-deVries (KdV) equation [26],

$$\partial_t f + f\partial_x f + \delta^2 \partial_{3x} f = 0. \tag{169}$$

Putting $u = f$ and $g = -\delta^2 \partial_{3x} f$ in the source and the branch equations for the hyperbolic equation, we can solve equation (169) numerically. We have used the space-centered finite difference equation to obtain the values of $\partial_{3x} f$ in the KdV equation. Figure 10 shows the typical numerical results for the soliton wave after 5000 time steps, where (a), (b), (c), and (d) are the results, respectively, by the 1-D 2nd KOND-H scheme, the 1-D 1st KOND-Hα scheme, the 1-D 1st KOND-Hβ scheme, and the compact CIP scheme without interpolation check. In the figure, the numerical data of f_n and $\partial_x f_n$ are shown again by the ♦ marks together with the initial sine profile of $f(t, x)$ by the solid lines. It is seen from Figure 10a that the soliton wave

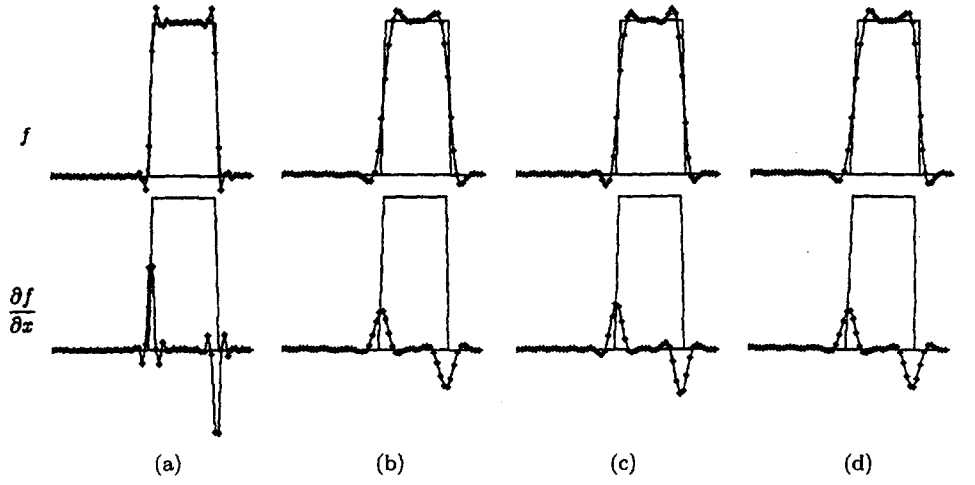

Figure 9. Typical results of computation for the linear wave propagation for the case of a square wave after 1000 time steps, k $(= c\Delta t/\Delta x)$ being 0.2. (a) 1-D 2nd KOND-H scheme, $\sigma = 0.027$. (b) 1-D 1st KOND-Hα scheme, $\sigma = 0.052$. (c) 1-D 1st KOND-Hβ scheme, $\sigma = 0.048$. (d) Compact CIP scheme without interpolation check, $\sigma = 0.052$. Numerical data of f_n and $\partial_x f_n$ are shown by the ♦ marks together with the analytical profile of $f(t, x)$ by the solid lines.

is well-realized with fairly high numerical stability in both the profiles of f_n and $\partial_x f_n$ by using the 1-D 2nd KOND-H scheme. In detailed observations, however, we find that there appears higher diffusion and noise in the calculation of higher derivatives of f in general, such as $\partial_{2x} f_n$, as was reported in the previous paper [19]. This is because we use the smaller number of terms in the Taylor expansion for the interpolation curves in the higher order derivatives, as is seen from equations (122)–(124). We recognize from comparison among the three Figures 10a, 10b and 10c that both the 1-D 1st KOND-Hα and the 1-D 1st KOND-Hβ schemes yield almost the same results of the soliton wave with those by the 1-D 2nd KOND-H scheme in both the profiles of f_n and $\partial_x f_n$. When we observe in detail the data of f_n at the highest peak of the soliton waves, we find that there takes higher numerical diffusion in the 1-D 1st KOND-Hα scheme of Figure 10b compared with the 1-D 2nd KOND-H scheme of Figure 10a, and the error by the numerical diffusion is improved in the 1-D 1st KOND-Hβ scheme of Figure 10c compared with the 1-D 1st KOND-Hα scheme. (These differences of the numerical accuracy will be shown quantitatively in the next figure.) On the other hand, it can be found from comparison between Figures 10b and 10d that the numerical diffusion takes place to suppress further the amplitudes of the soliton wave and there appears higher noise in $\partial_x f_n$ in the calculation by the compact CIP scheme of Figure 10d, compared with the 1-D 1st KOND-Hα scheme of Figure 10b. When we compare the numerical results of Figures 8b and 8d for the linear wave propagation with those of Figures 10b and 10d for the nonlinear wave propagation like as the soliton wave, we find that the 1-D 1st KOND-Hα scheme yields fairly accurate and stable numerical results compared with the compact CIP scheme for the case of the nonlinear process, while both of the two schemes yield similar numerical results for the linear wave propagation. This difference originates from the nearly analytical approximate solutions used for the construction of the 1-D 1st KOND-Hα scheme in order to suppress more effectively the loss of [Inf.1] on the "relations" in the partial differential equations, as was shown in detail in Section 4.1, compared with the compact CIP scheme [16–18]. The CPU times used for the computations of Figure 10 are about 131 sec by the 1-D 2nd KOND-H scheme, 70 sec by the 1-D 1st KOND-Hα scheme, 83 sec by the 1-D 1st KOND-Hβ scheme, and 55 sec by the compact CIP scheme.

In order to see the numerical error of the data by the three schemes of the 1-D 1st KOND-Hα, the 1-D 1st KOND-Hβ, and the compact CIP more quantitatively, we use here a relative root mean square deviation, σ_R, of the data f_n^k by the three schemes from the data $f_n^k(2^{\text{nd}})$ by the

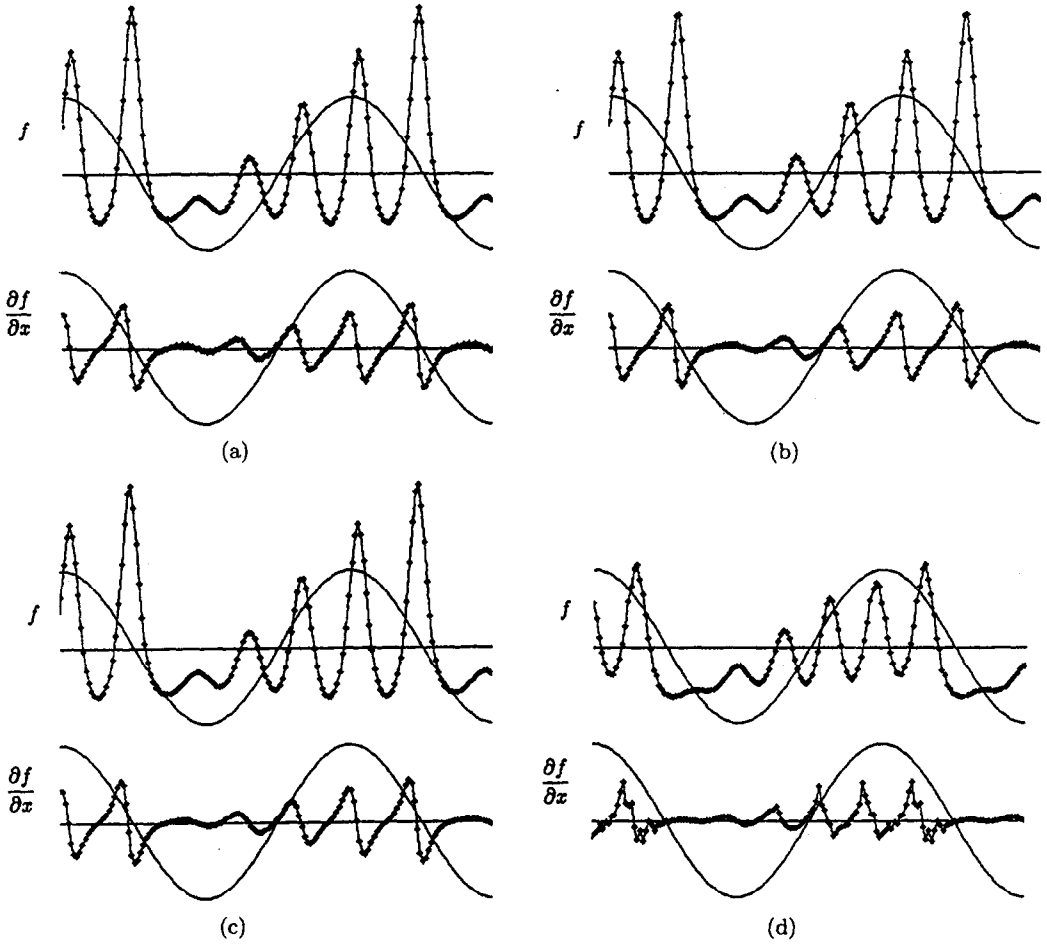

Figure 10. Typical results of computation for the nonlinear wave propagation for the case of the soliton wave after 5000 time steps. (a) 1-D 2nd KOND-H scheme, CPU time = 131 sec. (b) 1-D 1st KOND-Hα scheme, CPU time = 70 sec. (c) 1-D 1st KOND-Hβ scheme, CPU time = 83 sec. (d) Compact CIP scheme without interpolation check, CPU time = 55 sec. Numerical data of f_n and $\partial_x f_n$ are shown by the ♦ marks together with the initial sine profile of f by the solid lines. $\Delta t/\Delta x = 0.1/2.0$. $\delta = 1.8$. The amplitude and the wave length of the initial sine profile of f are 0.2 and 200, respectively.

1-D 2nd KOND-H scheme, where σ_R is defined by

$$\sigma_R = \left\{ \frac{1}{N} \sum_{n=1}^{N} \left[f_n^k - f_n^k \left(2^{\text{nd}}\right) \right]^2 \right\}^{1/2} . \tag{170}$$

Figure 11 shows typical time evolutions of the relative root mean square deviations, σ_R, of the data by the three schemes of the 1-D 1st KOND-Hα, the 1-D 1st KOND-Hβ, and the compact CIP, where the three marks of □, ■, and ♦ denote the numerical data, respectively, by the 1-D 1st KOND-Hα, the 1-D 1st KOND-Hβ and the compact CIP schemes. We see from Figure 11 that the relative numerical error by the 1-D 1st KOND-Hα scheme is smaller by over one order than that by the compact CIP scheme. The numerical accuracy by the 1-D 1st KOND-Hβ scheme is improved further compared with the 1-D 1st KOND-Hα scheme, as is shown in Figure 11. We recognize from the data by the two 1-D 1st KOND-H schemes in Figure 11 that the numerical error is kept to be small and constant until 1000 time steps, and it begins to increase at around 1200 time steps. On the other hand, the numerical error by the compact CIP scheme increases from the beginning of the data curve, as is shown in Figure 11. In order to investigate relations between wave forms and the increase of the numerical error, Figure 12 shows three temporal wave forms of f_n^k at 1000, 1200, and 1400 time steps, which are obtained by the 1-D 2nd KOND-H

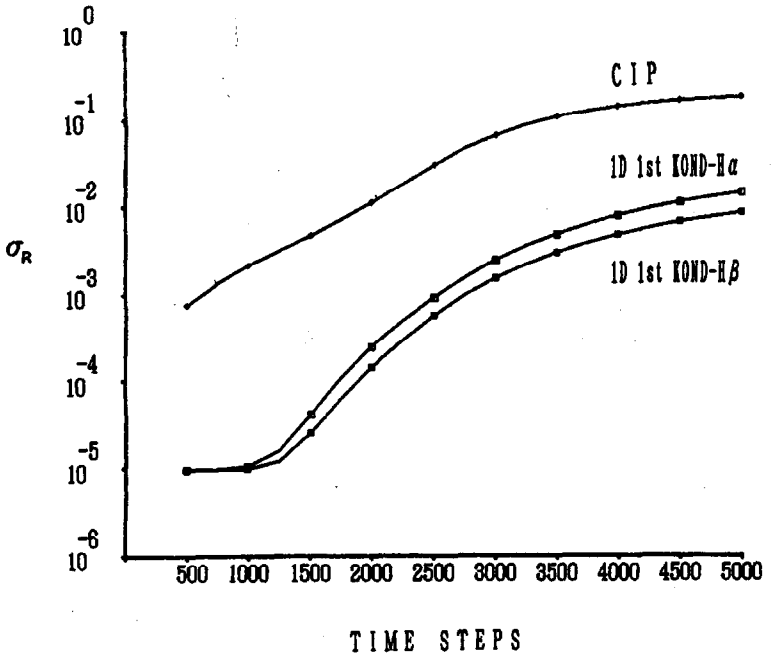

Figure 11. Typical results of the time evolution of the relative root mean square deviations, σ_R, from the data by the 1-D 2nd KOND-H scheme for the computation of the soliton wave. Three lines of σ_R for the 1-D 1st KOND-Hα scheme (the mark □), the 1-D 1st KOND-Hβ scheme (the mark ■), and the compact CIP scheme (the mark ♦) are shown in a semi-log scale. $\Delta t/\Delta x = 0.1/2.0$. $\delta = 1.8$. The amplitude and the wave length of the initial sine profile of f are 0.2 and 200, respectively.

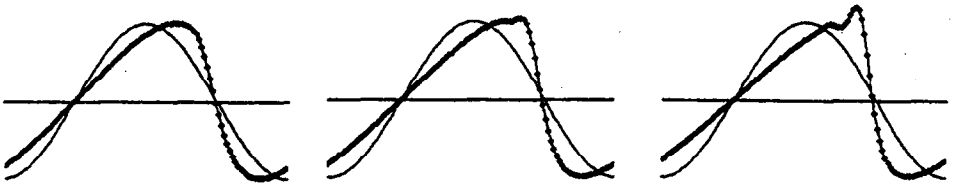

Figure 12. Temporal wave forms of f_n^k, obtained by the 1-D 2nd KOND-H scheme. (a) 1000 time steps. (b) 1200 time steps. (c) 1400 time steps. $\Delta t/\Delta x = 0.1/2.0$. $\delta = 1.8$. The amplitude and the wave length of the initial sine profile of f are 0.2 and 200, respectively.

scheme. We find from the comparison between the wave forms of Figure 12 and the time evolution of the numerical error by the two 1-D 1st KOND-H schemes in Figure 11 that the numerical error is kept to be small and constant while the wave form has moderate profiles, and it begins to increase at around 1200 time steps when the wave form begins to have sharp peaks. This feature can be understood to originate from the fact that the second branch equation for the second derivatives $\partial_{2x} f(t,x)$ is not solved in the two 1-D 1st KOND-H schemes, while it is solved in the 1-D 2nd KOND-H scheme. On the other hand, the numerical error by the compact CIP scheme increases even if the wave form has moderate profiles such as the wave form at 1000 time steps, as is seen from the comparison between Figure 11 and Figure 12. This may be understood to originate from the higher loss of [Inf.1] in the compact CIP scheme, compared with the two 1-D 1st and the 1-D 2nd KOND-H schemes where the loss of [Inf.1] is suppressed by using the nearly-analytic, higher order approximate solutions.

5. DISCUSSION AND SUMMARY

We have presented in detail two applications of the KOND algorithm to the parabolic and the hyperbolic type equations. In the data processing for numerical schemes to solve differential equations, we depend upon the two types of information, i.e., [Inf.1] on the "relations" and [Inf.2] on the functional "values," as was discussed in Section 2 for the thought analysis on numerical schemes. When we map the information of [Inf.2] onto the grid points, they are expressed by the infinite set of the discrete values $\{f_n, \partial_x f_n, \partial_{2x} f_n, \ldots\}$ of equation (6), based upon the interpolation curves by the Taylor expansions. The information of [Inf.1] is also mapped onto the infinite set of the discrete values on the grid points, and they are expressed by the infinite set of the relations of equation (3), i.e., the source and their branch equations. Since we cannot use all elements of these two infinite sets of equations (3) and (6), we have to adopt finite elements from the lower order derivatives. We then inevitably lose a large part of the information of both [Inf.1] and [Inf.2], and therefore, numerical errors accumulate gradually during the data processing. The thought analysis on the numerical schemes to attain higher numerical accuracy leads inevitably to the KOND algorithm with the four main elements of thoughts $\{[\mathrm{I}], [\mathrm{II}], [\mathrm{III}], [\mathrm{IV}]\}$ of equation (14). All of the four main thoughts $\{[\mathrm{I}], [\mathrm{II}], [\mathrm{III}], [\mathrm{IV}]\}$ are necessary to suppress effectively more the losses of [Inf.1] and [Inf.2] in order to attain higher numerical accuracy, as was discussed in Section 2.

In the KOND algorithm presented here, we have confined ourselves to problems possessing the requisite level of continuity. When we treat problems possessing discontinuities, we would have to introduce some special ideas in the data processing for local regions containing singular points $\mathbf{x}_s$ where discontinuity takes place. Here, the singular points $\mathbf{x}_s$ denote singular points in one-dimensional problems, singular lines in two-dimensional ones, or singular surfaces in three-dimensional ones. The singular points $\mathbf{x}_s$ are considered to be characterized by values of $\{[f(\mathbf{x}_s)_{-0}, f(\mathbf{x}_s)_{+0}], [\partial_i f(\mathbf{x}_s)_{-0}, \partial_i f(\mathbf{x}_s)_{+0}], [\partial_{ij} f(\mathbf{x}_s)_{-0}, \partial_{ij} f(\mathbf{x}_s)_{+0}], \ldots\}$ together with their positions $\mathbf{x}_s$. (We note that regular points with "continuity" are those where $\{[f(\mathbf{x})_{-0} = f(\mathbf{x})_{+0}], [\partial_i f(\mathbf{x})_{-0} = \partial_i f(\mathbf{x})_{+0}], [\partial_{ij} f(\mathbf{x})_{-0} = \partial_{ij} f(\mathbf{x})_{+0}], \ldots\}$.) As one of the useful methods to treat the singular points, one of the authors (Y. K.) and his coworkers had proposed the local multisubscales and delta function (LMS-DF) method in [25]. In the LMS-DF method, multisubscales with finer grid points are locally opened around the singular points in addition to the mainscale grid points, and the singular points are traced and processed by using approximate local analytic solutions and a delta functionlike treatment [25]. The LMS-DF method is applied to the compact CIP scheme in reference [25] to demonstrate quite high numerical accuracy for a square wave propagation with $\sigma = 0.00068$ after 1000 time steps, while the compact CIP scheme without the LMS-DF method yields $\sigma = 0.052$, as was shown at Figure 9d. This LMS-DF method is also applicable to the present KOND algorithm because of its structural relation with the host numerical algorithm [25, Figure 2]. Other excellent ideas of the discontinuity detection developed by A. Harten for his ENO methods with subcell resolution [27] would be also applicable to the present KOND algorithm in order to suppress oscillations around the singular points $\mathbf{x}_s$ and to attain higher numerical accuracy. A new method to treat a sharp discontinuity by the density function proposed recently in [28] would be another useful idea applicable to the present KOND algorithm.

We have presented in detail the KOND algorithm for the parabolic equations in Section 3 and have shown the two schemes of the 1-D 1st KOND-P scheme and the 2-D 1st KOND-P one. One of the important procedures for suppressing the loss of [Inf.1] in these KOND-P schemes is that the branch equations up to the third order are used to construct the numerical schemes, as shown at equations (26)–(28) and equations (60)–(65). We have demonstrated the high numerical accuracy of the KOND-P schemes by the typical numerical results which show less numerical errors than the conventional explicit scheme by over 2–3 orders, measured quantitatively by the root mean square deviation from the analytical solution, as are shown in Figures 2 and 6. We have also shown

that the 2-D 1st KOND-P scheme yields the same numerical accuracy by using only about $\frac{1}{5}$ of the CPU time used by the conventional explicit scheme of reference, as was shown at Figures 6 and 7. This result indicates that numerical schemes with quite high accuracy lead to effective high reduction of the CPU time to attain the same common numerical accuracy.

We have presented in detail the KOND algorithm for the hyperbolic equations in Section 4 and have shown the three schemes of the 1-D 2nd KOND-H, the 1-D 1st KOND-Hα, and the 1-D 1st KOND-Hβ. We have demonstrated numerically in Figures 8–11 that all of the three KOND-H schemes yield much less diffusive error and has fairly high stability for both of the linear and the nonlinear wave propagations, compared with the compact CIP scheme [15–18] which is known to be less diffusive compared with other conventional schemes such as FCT [6], QUICKEST [7], TVD [10,11], and PPM [12,13]. The 1-D 2nd KOND-H scheme solving up to the 2nd derivatives yields the highest numerical accuracy compared with the other two 1-D 1st KOND-H schemes solving up to the 1st derivatives, and the difference of the numerical accuracy between the 1-D 2nd and the 1-D 1st KOND-H schemes indicates the importance of the two thoughts of $\{[\,\mathrm{I}\,], [\,\mathrm{II}\,]\}$ in equation (14) in order to suppress the loss of [Inf.1] to attain the higher accuracy (Figures 8, 9, and 11). The 1-D 1st KOND-Hβ scheme with three connection relations yields higher numerical accuracy than the 1-D 1st KOND-Hα scheme with two connection relations, and the difference of the numerical accuracy between these two 1-D 1st KOND-H schemes shows the importance of the thought of [IV] in equation (14) in order to suppress the loss of [Inf.2] to attain the higher accuracy (Figures 8 and 11).

We have shown and discussed the origins of numerical errors by using typical numerical results and connecting the losses of [Inf.1] and [Inf.2]. The KOND algorithm would seem to be a somewhat abstract algorithm to attain higher numerical accuracy by suppressing the losses of [Inf.1] and [Inf.2]. We have presented, however, the KOND algorithm in detail in Section 3 for the parabolic type equations and in Section 4 for the hyperbolic type equations, by showing the procedures, step by step, for each thought element of the set $\{[\,\mathrm{I}\,], [\,\mathrm{II}\,], [\mathrm{III}], [\mathrm{IV}]\}$ of equation (14). We believe that the KOND algorithm would be useful to construct novel numerical schemes with higher numerical accuracy, when we investigate further the finer structure of the problem being studied, as was shown and demonstrated in the present paper.

REFERENCES

1. D.W. Peaceman and H.H. Rachford, Jr., *J. Soc. Indust. Appl. Math.* **3**, 28 (1955).
2. J. Douglas, Jr., *J. Soc. Indust. Appl. Math.* **3**, 42 (1955).
3. P.D. Lax and B. Wendroff, *Comm. Pure Appl. Math.* **13**, 217 (1960).
4. R. Richtmyer and K. Morton, *Difference Methods for Initial-Value Problems*, Interscience, New York, (1967).
5. J.E. Fromm, *J. Comput. Phys.* **3**, 176 (1968).
6. D.L. Book, J.P. Boris and K. Hain, *J. Comput. Phys.* **18**, 248 (1975).
7. B.P. Leonard, *Comp. Meth. Appl. Mech. Engng.* **19**, 59 (1979).
8. A. Brandt, *AIAA J.* **18**, 1165 (1980).
9. J.F. Thompson, Z.U.A. Warsi and C.W. Mastin, *J. Comput. Phys.* **47**, 1 (1982).
10. A. Harten, *SIAM J. Numer. Anal.* **21**, 1 (1984).
11. H.C. Yee, NASA Report, USA, TM-89464, (1987).
12. P. Colella and P.R. Woodward, *J. Comput. Phys.* **54**, 174 (1984).
13. J.B. Bell, C.N. Dawson and G.R. Shubin, *J. Comput. Phys.* **74**, 1 (1988).
14. E. Livne and A. Glasner, A finite difference scheme for the heat conduction equation, *J. Comput. Phys.* **58**, 59 (1985).
15. H. Takewaki, A. Nishiguchi and T. Yabe, *J. Comput. Phys.* **61**, 261 (1985).
16. T. Yabe and E. Takei, *J. Phys. Soc. Jpn.* **57**, 2598 (1988).
17. T. Yabe and T. Aoki, *Comput. Phys. Commun.* **66**, 219 (1991).
18. T. Yabe, T. Ishikawa, P.Y. Wang, T. Aoki, Y. Kadota and F. Ikeda, *Comput. Phys. Commun.* **66**, 233 (1991).
19. Y. Kondoh, *J. Phys. Soc. Jpn.* **60**, 2851 (1991).
20. Y. Kondoh, A physical thought analysis for Maxwell's electromagnetic fundamental equations, *Rep. Electromagnetic Theory Meeting of IEE Japan*, 1972, EMT-72-18 (in Japanese).
21. Y. Kondoh, Thought analysis on relaxation and general principle to find relaxed state, Research Rep., National Institute for Fusion Science, NIFS-109, Nagoya, Japan, (1991).

22. Y. Kondoh, Internal structures of self-organized relaxed states and self-similar decay phase, Research Rep., National Institute for Fusion Science, NIFS-141, Nagoya, Japan, (1992).
23. Y. Kondoh and T. Sato, Thought analysis on self-organization theories of MHD plasmas, Research Rep., National Institute for Fusion Science, NIFS-164, Nagoya, Japan, (1992).
24. Y. Kondoh and Y. Hosaka, Kernel optimum nearly-analytical discretization (KOND) method applied to parabolic equation (KOND-P Scheme), Research Rep., National Institute for Fusion Science, NIFS-118, Nagoya, Japan, (1991).
25. Y. Kondoh, J.L. Liang, T. Yabe, T. Ishikawa and S. Yamaguchi, *J. Phys. Soc. Jpn.* **59**, 3033 (1990).
26. D.J. Korteweg and G. de Vries, *Philos. Mag. Ser.* **5, 39**, 422 (1895).
27. A. Harten, *J. Comput. Phys.* **83**, 143 (1989).
28. T. Yabe and F. Xiao, Tracking sharp interface of two fluids by the CIP (Cubic-Interpolated Propagation) scheme, Research Rep., National Institute for Fusion Science, NIFS-206, Nagoya, Japan, (1993).